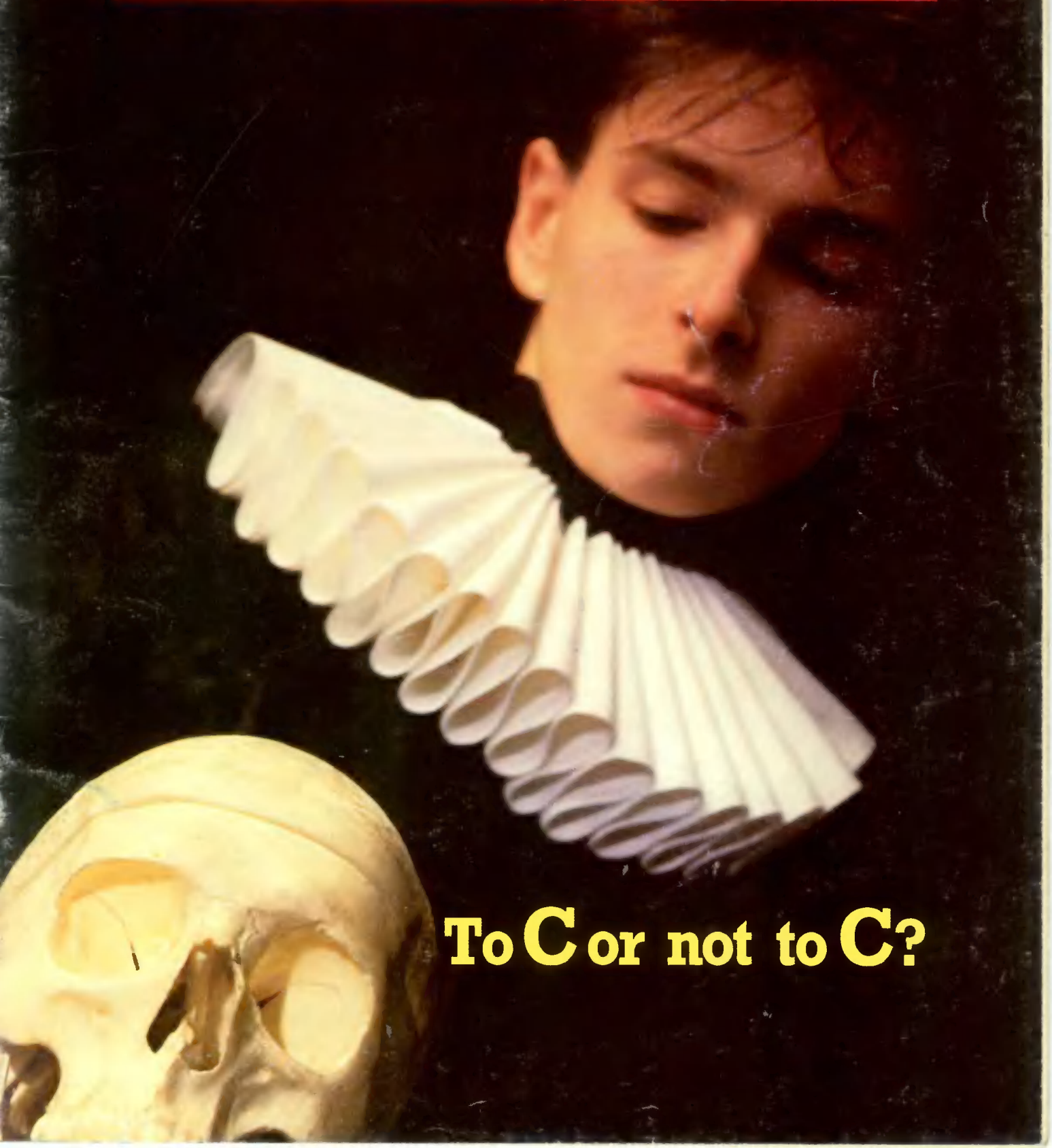
NOWY IDOL Z BORLANDU

mikroklan

Magazyn informatyczny HT ISSN 0860-1941 SIGMA cena zł.300 listopad-grudzień '87



To C or not to C?

nowy mikrokomputer
LIDIA K&K

kosztuje
30%
taniej

pracuje
50%
szybciej

daje
100%
pewności

że zmieni Wasze zdanie
o mikrokomputerach krajowej produkcji

oferujemy:

systemy wielodostępne i sieciowe
oprogramowanie użytkowe i narzędziowe

firmy
Computer Studio Kajkowscy

Prowadzimy wdrożenia i szkolenia zarówno na miejscu jak i u klienta

ul. Balladyny 15, PL 81-524 Gdynia, ☎ (0-58) 29 00 18, 054792 csk pl

EO 484-87 Q

Numer z podwójną datą – listopad-grudzień – kończy rocznik 1987. Następny w kolejności ukaże się **mikroklan** 1(13) styczeń 1988, poświęcony w dużej części zapowiadanej wcześniej tematyce Desktop Publishing.
Pierwszy przyszłoroczny numer podpisze do druku samodzielna już redakcja magazynu informatycznego **mikroklan**. Po okresie redagowania czasopisma w zespole „Informatyki" (7 numerów), a następnie w zespole „Horyzontów Techniki" (5 numerów), będzie ono redagowane w zespole... **mikroklan**u, koncentrującym się bez reszty na tworzeniu nowoczesnego miesięcznika o tematyce komputerowej. Jak zawsze pomocna będzie przy tym współpraca Czytelników i Autorów z redakcją. Ze swej strony dziękuję jednym i drugim za przekazane w drugim półroczu br. uwagi o czasopiśmie i materiały do publikacji.

Tadeusz Rathman

Redaguje zespół ***Horyzontów Techniki***
Redaktor naczelny - Tadeusz Rathman,
z-ca red. nacz. - Andrzej J. Piotrowski (**mikroklan**),
z-ca red. nacz. - Piotr Czarnowski, sekretarz redakcji - Jerzy Wierzbowski,
z-ca sekr. red. - Alicja Wancerz-Gluza, redaktorzy: Tomasz Dec, Krzysztof Kamiński, Maciej A. Markowski, Sławomir Piotrowski, Krzysztof Rzymkowski, Romuald Szuniewicz, Janusz Wrześniak.
Stali współpracownicy: Ryszard Rzędkowski, Andrzej Suwała .
Prace wydawnicze: Jan Tuszyński.
Komputerowe przygotowanie tekstów: Metrum - Krystyna W. Piotrowska.
Prace techniczne Roland Hojczak.
Sekretariat - Anna Graczyk.
Adres redakcji: ul. Swiętokrzyska 14a, PL 00-950 Warszawa, skrytka 1004.
Telefony: 27-26-08, 27-47-37, 26-74-61 w.473.
Akwizytor ogłoszeń Zbigniew Rutkowski, tel.33-62-77, 26-20-27.
Wydawca: Wydawnictwo Czasopism i Książek Technicznych ***SIGMA,***
Przedsiębiorstwo Naczelnej Organizacji Technicznej.
Skład: WZKart (Linotronic 300)–operator systemu Piotr Kurzec.
Druk: Bohmann Druck und Verlag GmbH co. KG Wien, Austria.
Druk na zlecenie Ars Polona.
Nakład 100 000 egz. K-65 ISNN 0860-1941 Cena 300 zł.
Stałe otrzymywanie **magazynu informatycznego *Ht* mikroklan** można sobie zapewnić wysyłając zamówienie bezpośrednio do Zakładu Kolportażu Wydawnictwa NOT-SIGMA ul. Bartycka 20, 00-950 Warszawa, skrytka 1004. Zamówienia można składać na wybrane numery oraz okresy kwartalne, półroczne i roczne w ramach danego roku kalendarzowego. Kontynuacja otrzymywania czasopisma w roku następnym wymaga ponownego złożenia zamówienia. Zamówienia będą realizowane w kolejności wpływu, aż do wyczerpania nakładu. Zamawiający nie dokonują przedpłat. Instytucje podają w zamówieniu warunki płatności. Odbiorcy indywidualni otrzymują zamówione egzemplarze za zaliczeniem pocztowym równym cenie czasopisma (porto opłaca nadawca). Informacji udziela Dział Handlowy, tel. W-wa, 40-00-21 w. 207, 238 lub 40-37-31.

mikroklan 9(12) listopad-grudzień 1987

Turbo C dobry na wszystko?
To C or not to C? 2

Szybciej krócej, taniej... 2

Kontrowersje 4

System Turbo C 4

Ściągawka mikroklanu 5

Komputer wszystko Ci powie...
Mowa syntetyczna 7

Szybko i przejrzyście
elektroniczne arkusze obliczeniowe
1-2-3 10

Standardy modułowych systemów automatyki
Camac i inne 12

Sprzęt z pozycji programisty
BIOS IBM PC 15
Zestawienie procedur BIOS 16-17

Programiści w defensywie
Zabezpieczanie programów 19

Bez bandaża, jodyny i plastra...
Ratujmy nasze zbiory 21

Tym razem o dyskietkach
MS-DOS od środka (5) 23

Koputer robi komputer... i nie tylko
Płytka z komputera 26
Smartwork 26
Platine 64 27

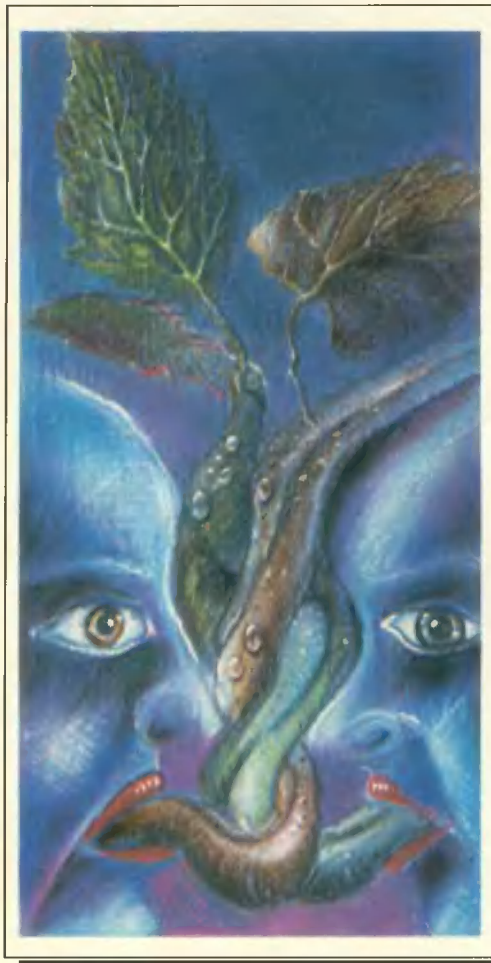
Nowe oblicze poczty
Teletex 28

Kuchnia komputerowej grafiki
Trzeci wymiar (1) 29

Co czytać (po angielsku)
Elementarz dla profesjonalistów 31
Spis treści 1986-1987 32
mikrokram 7, 11, 14, 25, 28
Galeria mikroklanowa 9

Przepraszamy p. Sławomira Jagiełłę — współautora artykułu pt. Płyta główna PC XT, drukowanego w **mikroklanie** 7/87 – za pominięcie Jego nazwiska. (red)

To C or not to C?



Język C powstał ok. 1972 r. w Bell Laboratories w USA. Od tego czasu zyskał sobie wielu zwolenników wśród zawodowych programistów. Opracowano w nim wiele programów użytkowych (programy do obliczeń tablicowych, bazy danych, procesory tekstu itd.) i bogate oprogramowanie systemowe (koronnym przykładem jest tu system operacyjny UNIX). Co szczególnego jest w języku C, że zyskuje on taką popularność?
Do najważniejszych zalet języka C należy zaliczyć:
- elastyczność (nie jest to język specjalizowany),
- zwarty kod i znaczną szybkość realizacji programów,
- możliwość stosowania go na różnych komputerach (od ZX Spectrum, poprzez mikrokomputery pracujące pod kontrolą CP/M, IBM PC XT/AT, VAX, skończywszy na dużych systemach komputerowych) - bez konieczności wprowadzania istotnych modyfikacji w programach źródłowych,
- zapewnienie dostępu do mechanizmów sprzętowych komputera,
- możliwość strukturalnego zapisu algorytmu.

Język C tworzony był z myślą o wykorzystaniu go do pisania oprogramowania systemowego. Doskonale sprawdził się przy systemie UNIX - tylko niewielkie fragmenty systemu musiały zostać zapisane w języku maszynowym. Wzrastająca popularność UNIX-a (i jego pochodnych np. XENIX, QNX) pociągnęła za sobą zwiększone zainteresowanie językiem C. Mimo niewątpliwych zalet, popularność C, wynika w dużej mierze ... z mody. Propagując język C na łamach **mikro**klanu nie zamierzamy nikogo namawiać, by zaczynał naukę programowania właśnie od niego. Do nauki lepszy jest Pascal, który zapewnia większą przejrzystość zapisów - jest to bardzo istotne przy stawianiu pierwszych kroków. Nie będziemy też publikować kursu języka C, choć bez wątpienia **mikro**klan zyskałby dzięki temu większą popularność. Sądzimy, że po chwili zastanowienia Czytelnicy przyznają nam rację. Nie sposób omówić języka C (czy też jakiegokolwiek innego nowoczesnego języka) w kilku krótkich publikacjach. Polecamy natomiast wykłady i szkolenia (jest ich coraz więcej, nawet w mniejszych ośrodkach), a samoukom - książki (PWN niedawno wydało dwie i zapowiada trzecią). Publikacje o języku C, które zamieszczamy na łamach **mikro**klanu, mają służyć doskonaleniu umiejętności, wskazywać metody i ... braki w niektórych rozwiązaniach.

mikroklan

Turbo C

Szybciej, krócej, taniej...

Firma Borland ceniona jest za produkty o dobrych parametrach i niskiej cenie. Dobre nie oznacza oczywiście „najlepsze", ale zaspokajające potrzeby możliwie szerokich kręgów odbiorców. Do najbardziej znanych produktów należy Turbo Pascal. Borland systematycznie rozszerza swoją ofertę: wprowadził Turbo Prolog, Turbo Basic (mikro**klan 8/87) i niedawno Turbo C, który ma wielu konkurentów, ale bez wątpienia bije ich niską ceną.**

W nowoczesnych programach użytkownik zwolniony jest z konieczności zapamiętywania dyrektyw dzięki rozbudowanym menu, w których wymienione są możliwe w danej sytuacji operacje - wystarczy wybrać jedną z nich. Firma Borland zastosowała metodę menu już w Turbo Pascalu, jednak zakres oferowanej w ten sposób pomocy był stosunkowo ograniczony. Popularna od pewnego czasu technika rozwijanych menu została natomiast zastosowana przez Borland w Turbo Prologu, a także w Turbo C. Rozwijane menu pozwala skrócić czas zapoznawania się z programem, jego możliwościami i opcjami. Nie bez znaczenia jest też fakt, że przypomina użytkownikowi o konieczności konkretnych wyborów (np. czy program ma odwoływać się do koprocesora arytmetycznego czy też emulować jego funkcje).
Przeciwnicy techniki rozwijanych menu (a jest ich coraz mniej) twierdzą, że stanowią one rodzaj warstwy izolacyjnej: wpisanie komendy trwa krócej niż przebrnięcie przez poszczególne menu. Dlatego też w pakiecie Turbo-C umieszczony został też drugi, sterowany komendami kompilator przeznaczony dla profesjonalistów.

Potrzeby

System Turbo-C teoretycznie może być wykorzystany nawet w komputerach o minimalnej konfiguracji (tzn. z pojedynczym napędem dysków elastycznych). Nie ma to jednak większego sensu. Składające się na system zbiory zajmują bowiem ponad 230 KB. Do tego trzeba jeszcze doliczyć zbiory biblioteczne i okaże się, że na uruchamiany program nie ma już miejsca.
Pamięć operacyjna komputera musi mieć więc co najmniej 384 KB. W zasadzie żądanie to można spełnić bez problemów, gdyż komputery klasy PC bardzo rzadko oferowane są z pamięcią RAM mniejszą od 512 KB. Kłopoty mogą jednak wystąpić w przypadku stosowania programów rezydujących w pamięci operacyjnej.

Możliwości

Autorzy Turbo-C postanowili zapewnić efektywne wykorzystanie sprzętu. Dlatego też użytkownik może np. wybrać typ mikroprocesora (8088 lub 80186/286). Dzięki czemu programy realizowane na komputerach PC AT będą miały krótszy czas wykonania nie tylko ze względu na większą moc przetwarzania jednostki centralnej, ale też ze względu na optymalny kod powstający w wyniku kompilacji. Efektywne gospodarowanie pamięcią komputera umożliwia

wprowadzenie sześciu modeli pamięci, w tym rzadko spotykanego modelu miniaturowego (ang. tiny).
Bezsporną wadą pakietu Turbo-C jest brak możliwości efektywnego wykorzystania mikroprocesora 80386, 32-bitowe modele PC znajdują coraz więcej odbiorców, a w nowej rodzinie PS/2 najlepsze perspektywy ma Model 80.
Na pochwałę zasługuje natomiast wyjątkowo efektywna obsługa pamięci masowych. W stosunku do systemów Microsoft-C i Lattice-C widoczna jest wyraźna różnica w czasie dostępu do zbiorów umieszczonych na dyskietce. Wynika to jednak z faktu, że Turbo-C nie umieszcza otrzymanego w wyniku kompilacji kodu bezpośrednio w pamięci operacyjnej (tak jak to było możliwe w Turbo Pascalu). Każdorazowe wykonanie programu pod kontrolą pakietu Turbo-C wymaga ponownego wprowadzenia zapisanego na dysku kodu do pamięci operacyjnej systemu.
W bibliotece Turbo-C do dyspozycji użytkownika znajduje się ponad 300 funkcji. Wśród nich należy wyróżnić funkcje umożliwiające wykorzystanie koprocesora arytmetycznego 8087 (80287) lub zastąpienie go procedurami w standardzie IEEE dla liczb zmiennoprzecinkowych. W funkcjach bibliotecznych zawarto też procedury pozwalające odwoływać się do funkcji DOS i BIOS oraz realizujące operacje wejścia-wyjścia.
W pakiecie przewidziano opcję „Make", która pozwala w programie głównym wykonywać programy biblioteczne bez potrzeby ponownej ich kompilacji. Modyfikowany jest jedynie kod wynikowy.
Mimo że programy kompilowane za pomocą Turbo-C cechuje stosukowo zwarty kod wynikowy i krótki czas wykonania, nie mogą one konkurować z programami przygotowanymi bezpośrednio w języku maszynowym. Turbo-C pozwala jednak zastosować rozkazy języka maszynowego w tekście źródłowym programu. Opcja ta przeznaczona jest do wykorzystania w przypadku, gdy w programie są procedury wymagające szczególnie krótkiego czasu wykonania lub odwołania się do układów sprzętowych z pominięciem standardowych funkcji.
W procesie uruchamiania programów istotna jest pomoc, jaką program może zaoferować przy wyszukiwaniu błędów. Pakiet Turbo-C pozwala na wykorzystanie do tego celu funkcji Debug. Chociaż zapewniane przez program możliwości są nieporównywalnie gorsze od Microsoft-C, w wielu przypadkach mogą okazać się wystarczające, tym bardziej że bardzo dobrze została zrealizowana diagnostyka błędów syntaktycznych.

Tabela 1. Porównanie parametrów najpopularniejszych kompilatorów języka C.

	DeSmet-C	Lattice-C	Microsoft-C	Turbo-C
Program NIC	14 bajtów	14 bajtów	14 bajtów	14 bajtów
Kompilacja	30 s	48 s	80 s	5 s
Konsolidacja	22 s	45 s	45 s	31 s
.OBJ	25 bajtów	179 bajtów	276 bajtów	170 bajtów
.EXE	1536bajtów	2614bajtów	1986 bajtów	1418 bajtów
czas wykonania		..... praktycznie nie mierzalny		
Program PUSTY	224 bajty	224 bajty	224 bajty	224 bajty
Kompilacja	40 s	57 s	89 s	11 s
Konsolidacja	34 s	56 s	55 s	33 s
.OBJ	207 bajtów	474 bajtów	452 bajtów	368 bajtów
.EXE	7168 bajtów	7680 bajtów	6864 bajtów	6394 bajtów
czas wykonania	6,2 s	7,5 s	3,6 s	3,6 s
Program SITO	559 bajtów	559 bajtów	559 bajtów	559 bajtów
Kompilacja	43 s	60 s	97 s	13 s
Konsolidacja	34 s	59 s	57 s	33 s
.OBJ	356 bajtów	567 bajtów	556 bajtów	453 bajtów
.EXE	7168 bajtów	7744 bajtów	6944 bajtów	6458 bajtów
czas wykonania	14,3 s	13,2 s	12,9 s	7,8 s

Tabela 2. Porównanie parametrów różnych produktów firmy Borland.

	Turbo-Pascal	Turbo-Basic	Turbo-C
Program NIC	28 bajtów	17 bajtów	14 bajtów
Kompilacja i konsolidacja	5 s	9 s	36 s
.COM/.EXE	11433 bajty	28992 bajty	1418 bajtów
czas wykonania		.....praktycznie nie mierzalny.....	
Program PUSTY	181 bajtów	134 bajty	228 bajtów
kompilacja i konsolidacja	5 s	9 s	45 s
.COM/.EXE	11513 bajtów	29136 bajtów	6378 bajtów
czas wykonania	4,8 s	3,0 s	1,2 s
Program SITO	606 bajtów	477 bajtów	559 bajtów
Kompilacja i konsolidacja	6,4 s	10 s	46 s
.COM/.EXE	11683 bajty	29376 bajtów	6458 bajtów
czas wykonania	20,3 s	19,5 s	7,8 s

Porównania

Do porównania parametrów pakietu Turbo-C z innymi produktami wykorzystano trzy programy. Pierwszy z nich o nazwie NIC jest najkrótszym możliwym do zapisania w języku C programem. Nie realizuje on żadnej funkcji. Drugi, o nazwie PUSTY polega na wykonaniu 100 000-krotnej pętli. Trzeci, to typowo stosowany w testach program Sito Eratosthenesa. W tabeli 1 zestawiono wyniki testów opublikowane w [2]. Dla programu Sito odbiegają one nieco od informacji z reklam firmy Borland, co wynika z zastosowania innej wersji programu.
W sumie porównania wypadają zdecydowanie na korzyść pakietu Turbo-C. Wśród testów zabrakło jednak podobnego zestawienia dla długiego i skomplikowanego programu, dzięki czemu w Turbo-C mógł być stosowany model pamięci Tiny. Ponieważ kompilator firmy Borland jest jednoprzebiegowy (Microsoft-C wykorzystuje kompilator trójprzebiegowy) wynik porównania mógłby wypaść znacznie gorzej. O przodującej roli pakietu Turbo-C będzie można dopiero wyrokować po zestawieniu z nowymi produktami: Aztec-C firmy Manx (wersja 4), Quick-C i MS-C (wersja 5).

Interesujące może być jednak zestawienie Turbo-C z innymi produktami firmy Borland (tab. 4). W przypadku Turbo Pascala nie występuje typ danych „long integer" (32-bitowa liczba całkowita) i w związku z tym nie można zrealizować 100 000-krotnej pętli. Natomiast dla Turbo-Basica chociaż występuje wspomniany typ danej, posługiwanie się nim w przypadku programu Pusty prowadzi do czasów wykonania w granicach kilku minut. Bez wątpienia wskazuje to słabą stronę Turbo-Basica, ale nie upoważnia do uogólniania wniosków. Z tego względu program Pusty został zmodyfikowany: pętla wykonywana jest 50 000 razy, co pozwala na zastosowanie 16-bitowej zmiennej typu całkowitego.
W programach Turbo-Pascal i Turbo-Basic istnieje opcja umożliwiająca składowanie kodu wynikowego w pamięci operacyjnej komputera. Oczywiście, wykorzystanie w testach tej opcji pozostawiłoby Turbo-C bez szans. Założono więc, że wszystkie pakiety wpisują kod wynikowy na dysk. Mimo to czas kompilacji i konsolidacji jest znacznie krótszy dla Turbo-Pascala i Turbo-Basica. Turbo-C zapewnia natomiast znacznie bardziej zwarty kod wynikowy i krótszy czas wykonywania programu. W przypadku specyficznych zastosowań (np. sprawdzanie poprawności algorytmu) efektywniejszym narzędziem może okazać się Turbo-Pascal. Natomiast programy użytkowe lepiej realizować wykorzystując Turbo-C.

Andrzej J. Piotrowski

W następnym numerze artykuły Jana Bieleckiego o terminologii związanej z językiem C i o procedurach we-wy w Turbo C

Jan Bielecki

System Turbo C

Wraz z pojawieniem się kompilatora języka Turbo C, liczba znaczących implementacji języka C, przeznaczonych dla mikrokomputerów IBM PC i PS/2, przekroczyła 20. Świadczy to dobitnie o popularności i znaczeniu tego narzędzia programowania.
Ponieważ wszystkie przyszłe kompilatory języka C będą realizowane w wersji zgodnej ze standardem ANSI, nie od rzeczy będzie podać, że wymaganie to spełnia już Turbo C. Pod tym względem wyprzedza on takie implementacje jak Aztec i Microsoft, udostępniając użytkownikowi nie tylko wszystkie konstrukcje dotychczasowego języka wzorcowego, ale także liczne rozszerzenia standardu, jak: modele pamięci, środki do programowania w języku maszynowym i wygodne środowisko operacyjne.
Nie bez znaczenia jest także możliwość generowania kodu wynikowego przeznaczonego do wykonywania w wirtualnym trybie pracy mikroprocesora 80286. Pozwoli to na przygotowywanie programów przeznaczonych do stosowania pod nadzorem systemu operacyjnego OS/2.
Dotychczasowe doświadczenia z systemem Turbo C wskazują, że stanie się on znaczącym narzędziem programowania, konkurującym z kompilatorami powszechnie uznawanymi za profesjonalne.
Ponieważ wnikliwa ocena systemu Turbo C będzie miała sens dopiero wtedy, gdy pojawią się inne produkty w pełni realizujące standard ANSI, należy z nią nieco zaczekać, bacznie obserwując co robi konkurencja firmy Borland. Jednak już obecnie można wyrazić pogląd, że rola systemu Turbo C w rozwoju języka C będzie zbliżona do tej, jaką w rozwoju języka Pascal odegrał system Turbo Pascal.

Jan Bielecki

Nie wszyscy obdarzają zaufaniem wczesne wersje programów. Nowy kompilator języka C firmy Borland International, nie jest - podobnie jak inne kompilatory - wolny od błędów. Jeżeli niebawem pojawi się poprawiona wersja Turbo C ma ona szansę wyprzeć z wielu obszarów zastosowań takie sławy, jak Aztec C czy Microsoft C.
Do zalet Turbo C należy duża szybkość kompilacji, konsolidacji i wykonania utworzonych za jego pomocą programów. Jednak w reklamach firmy Borland zestawiono Turbo-C ze starszymi wersjami kompilatorów. Nie ulega wątpliwości, że nawet w nowszej wersji kompilator firmy Microsoft nie pobije Borlanda w zakresie czasu kompilacji (szczególnie w przypadku krótkich programów - Turbo C jest kompilatorem jednoprzebiegowym). Większy będzie też kod wynikowy krótkich programów skompilowanych przy użyciu Microsoft-C (zaleta miniaturowego modelu pamięci dostępnego w Turbo-C). Porównanie czasów wykonywania programu wynikowego wypada jednak na korzyść firmy Microsoft w przypadku najnowszej wersji kompilatora. Tak zwane Sito (25 iteracji) realizowane jest przez IBM PS/2 w ciągu 5,7 sek. przy użyciu Microsoft C wersja 4.0 i w 3,3 sek. dla wersji 5.0. Zestawiając te czasy z podawanymi w reklamach firmy Borland można wyliczyć, że najnowszy produkt firmy Microsoft pozwala uzyskać program wykonywany nieco szybciej niż w wypadku użycia Turbo-C.
Świetna diagnostyka błędów syntaktycznych nie zmienia faktu, że największą wadą Turbo-C jest brak programu uruchomieniowego (debuggera).

Maciej A. Markowski

Zintegrowane środowisko operacyjne nazywane w tej pracy systemem Turbo C, składa się z kompilatora języka C, edytora, konsolidatora i bibliotek. Zostało ono opracowane przez firmę Borland z przeznaczeniem dla mikrokomputerów zgodnych z IBM PC.
Największą wygodę posługiwania się systemem Turbo C zapewnia umieszczenie go na dysku sztywnym. Przy stosowaniu dysku sztywnego zaleca się utworzenie w katalogu głównym podkatalogu i wypełnienie go zbiorami pochodzącymi z dyskietek dystrybucyjnych.
W przypadku, gdy użycie systemu Turbo C ograniczy się do programów, których ani kod, ani dane nie przekroczą 64 KB pamięci, można posłużyć się procedurą wsadową, przytoczoną na wydruku 1. W założonym podkatalogu (tutaj: JB-TC) należy umieścić przynajmniej zbiory TC.EXE i TCHELP.TCH oraz podkatalogi INCLUDE
i LIB. Podkatalog INCLUDE powinien zawierać pliki *.H oraz podkatalog SYS, a LIB zbiory CO?.OBJ, EMU.LIB, MATH?.LIB i C?.LIB.
Występujący w przytoczonych nazwach znak „?" reprezentuje jednoliterowe oznaczenie modelu pamięci. W przypadku programów, których ani kod, ani dane nie zamują więcej niż 64 KB, wystarcza posłużenie się małym modelem (S-small), a więc zbiorami COS.OBJ, EMU.LIB, MATHS.LIB i CS.LIB.
Po wykonaniu przytoczonych czynności system Turbo C jest zainstalowany i gotowy do pracy. Przy takim skonfigurowaniu, wszystkie moduły programów w języku C, m. in. moduły wynikowe będą umieszczane w tym samym podkatalogu, w którym znajduje się zbiór TC.EXE. Przed uruchomieniem systemu Turbo C należy wybrać podkatalog JB-TC, a następnie wywołać program TC.

Posługiwanie się menu

System Turbo C wywoływany jest dyrektywą TC. Na wstępie na ekranie jest wyświetlane menu główne, a na jego tle informacja o numerze wersji systemu. Naciśnięcie dowolnego klawisza powoduje usunięcie tej informacji i odsłonięcie ekranu podzielonego na:
- wiersz menu - składający się z pól File, Edit, Run, Compile, Project, Options i Debug,
- okno edycji - zatytułowane Edit,
- okno komunikatów - zatytułowane Message,
- wiersz kluczy - zawierający opisy funkcji wybranych kluczy.

W każdej chwili pracy systemu Turbo C aktywny jest wiersz menu albo jedno z okien. Po uaktywnieniu okna następuje wyróżnienie słowa tytułującego okno. Następuje też zamiana pojedynczej linii, na której znajduje się tytuł, na linię podwójną. Jeśli jest aktywny wiersz menu, to tylko wyróżniane jest jedno z jego pól.
Wybór czynności proponowanej w głównym menu może odbywać się na kilka sposobów. Najprostszym jest wprowadzenie z klawiatury pierwszej litery pola (wymienionego w menu). Można też posłużyć się klawiszami sterującymi kursor, a po przemieszczeniu się na wybrane pole menu - nacisnąć klawisz Enter. W każdym z przypadków nastąpi wywołanie menu niższego poziomu. Wybór czynności określonej w menu niższego poziomu odbywa się w podobny sposób. W niektórych przypadkach pojawia się kolejne menu - należy wówczas dokonać następnego wyboru. W przypadku omyłkowego wyboru można cofnąć się na wyższy poziom menu - naciskając klawisz Esc. Naciśnięcie tego klawisza, gdy wybrane jest główne menu, powoduje aktywowanie ostatniego aktywnego okna. Taki sam efekt można także uzyskać z dowolnego poziomu menu, naciskając klawisz F6. Po wybraniu dowolnego menu niższego poziomu, można posłużyć się klawiszami oznaczonymi strzałkami (w lewo lub w prawo), w celu przywołania na ekran sąsiednich menu niższego poziomu. Dostęp do częściej używanych pozycji menu można przyśpieszyć za pomocą wybranych klawiszy funkcyjnych. Przyjęto, że niezależnie od kontekstu, naciśnięcie klawisza F10 powoduje przejście do głównego menu, a jednoczesne naciśnięcie klawisza Alt oraz pierwszej litery pola wymienionego w głównym menu powoduje wybranie menu niższego poziomu związanego z tym polem. Naciśnięcie klawisza F1 powoduje wyświetlenie informacji o możliwości posłużenia się systemem Turbo C w danej sytuacji.

Przygotowywanie programów

Jednym z elementów systemu Turbo C jest edytor ekranowy. Można go wywołać z poziomu głównego menu (klawisz E) albo z dowolnego innego kontekstu (Alt-E). Zasady posługiwania się nim są zbliżone do zasad posługiwania się edytorem pakietu SideKick oraz systemu Turbo Pascal. Przypominają one w znacznym stopniu zasady obowiązujące w edytorze WordStar, na którym wszystkie wymienione edytory są wzorowane (patrz ściągawka).
Jeśli wywołanie edytora odbyło się bez określenia nazwy zbioru, to w momencie zakończenia edycji zostanie zaproponowana nazwa NOFILE.C (jeśli zbiór ma zostać zapisany do pamięci masowej). Nazwę tę można zmienić na dowolną inną. wprowadzając z klawiatury znak spacji, a bezpośrednio po nim nazwę zbioru. W przypadku zakończenia edycji przez wywołanie głównego menu, utworzony zbiór źródłowy nie zostanie zapamiętany w pamięci masowej, może jednak zostać poddany kompilacji, a następnie - po skonsolidowaniu z bibliotekami - wykonany. Skompilowanie i wykonanie programu może odbywać się z głównego menu (klawisz R) albo z dowolnego innego kontekstu (Alt-R).

W celu utworzenia i wykonania programu z wydruku 2 powinny zostać podjęte następujące czynności:
- wywołanie głównego menu (klawisz F10);
- wybranie opcji File (klawisz F);
- wybranie opcji New (klawisz N);
- wprowadzenie tekstu programu;
- skompilowanie i wykonanie programu (klawisz Alt-R).

Zapamiętanie programu jako zbioru (np. o nazwie HELLO.C) zapewniają czynności:
- wybranie opcji File (klawisz Alt-F);
- wybranie opcji Save (klawisz S);
- określenie nazwy zbioru (spacja, a po niej nazwa HELLO.C).

Program zapamiętany w zbiorze dyskowym może być oczywiście modyfikowany. W tym celu należy:
- wybrać opcję File (klawisz Alt-F);
- wybrać opcję Load (klawisz L);
- określić nazwę zbioru (spacja, a po niej nazwa - tu: HELLO);
- dokonać edycji programu (np. zmiana napisu JanB na Jan Bielecki);
- zapamiętać nową wersję zbioru na dysku (Ctrl-K-S).

W celu wykonania zmienionego programu, znajdującego się wciąż jeszcze w oknie edytora należy wybrać opcję RUN (klawisz Alt-R).
Zakończenie pracy w systemie Turbo C następuje po wybraniu opcji File (klawisz Alt-F), a następnie opcji Quit (klawisz Q).

Kompilowanie i wykonywanie programów

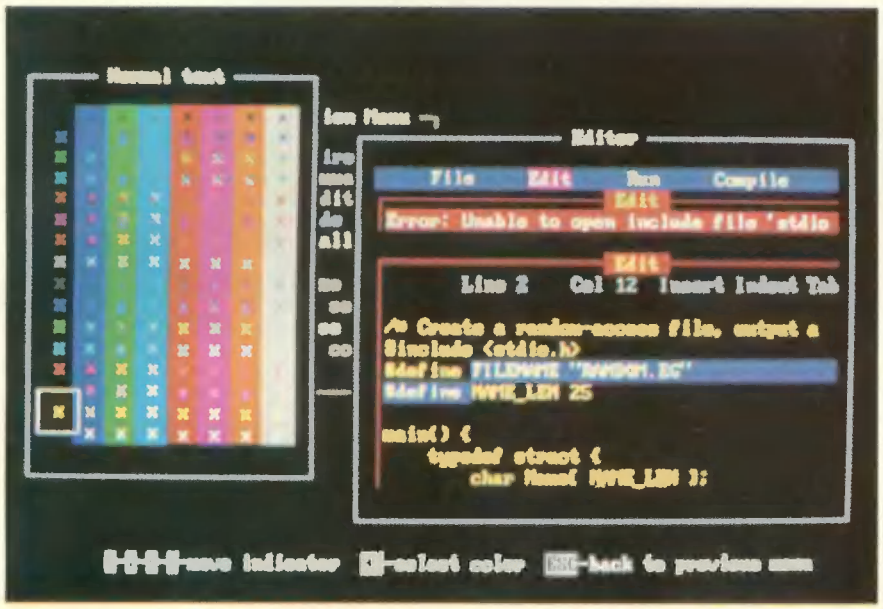
Kompilowaniu może podlegać zbiór znajdujący się w oknie edytora (klawisz Alt-F9) lub też jeden albo kilka zbiorów dyskowych (klawisz F9). W pierwszym przypadku nastąpi utworzenie modułu wynikowego z rozszerzeniem .OBJ - w drugim utworzenie programu wykonywanego z rozszerzeniem .EXE. Wybranie z głównego menu opcji Run wywołuje taki sam skutek, jak naciśnięcie klawisza F9 (make), a ponadto powoduje wykonanie utworzonego programu.

Literatura
1. Bielecki J.: Język C-interpretacja standardu, WNT 1987
2. Bielecki J.: Turbo C, Numerica 1987, Warszawa, ul. Alfreda Nobla 23D, tel.(0-22) 17-86-18

Obowiązują następujące ogólne zasady określania nazw zbiorów, których kompilacja wynika z naciśnięcia klawisza F9:
- jeśli uprzednio, w następstwie wybrania opcji Project, określono nazwę projektu (Project name), stanowiącą nazwę zbioru z rozszerzeniem .PRJ, to nastąpi skompilowanie wszystkich zbiorów źródłowych, których nazwy podano w tym zbiorze;
- jeśli nie określono nazwy projektu (tzn. w menu Project występuje nazwa pusta), ale w następstwie wybrania opcji Compile, określono tzw. nazwę zbioru pierwotnego (Primary C file) - nazwę z rozszerzeniem .C - to nastąpi kompilacja wykazanego zbioru;
- jeśli nie określono nazwy projektu, ani nazwy zbioru pierwotnego, to nastąpi kompilacja zbioru, który znajduje się w buforze edytora.
Bezpośrednio po zainstalowaniu systemu Turbo C, zarówno nazwa projektu, jak i nazwa zbioru pierwotnego jest pusta. Powoduje to, że kompilacji podlega zbiór źródłowy znajdujący się w oknie edytora. W przypadku prostych, jednomodułowych programów takie skonfigurowanie systemu jest w pełni zadowalające. Jeśli jednak program zawiera dyrektywy #include dotyczące zbiorów zawierających nie uruchomione jeszcze fragmenty programu, celowe jest określenie nazwy zbioru pierwotnego. Nazwa projektu musi zostać określona w przypadku programów wielomodułowych.

Przykład

Wykonanie programu jednomodułowego zawartego w pliku HELLO.C wymaga:
- wybrania opcji File (klawisz Alt-F);
- wybrania opcji Load (klawisz L);
- określenia nazwy zbioru (spacja, a po niej nazwa zbioru HELLO.C);
- wybrania opcji Run (klawisz Alt-R).
Wykonanie programu jednomodułowego zawartego w zbiorach (wydruk 3) niezależnie od tego, który ze zbiorów jest właśnie załadowany do edycji, wymaga:
- wybrania opcji Compile (klawisz Alt-C);
- wybrania opcji Primary (klawisz P) w celu określenia nazwy zbioru pierwotnego;
- określenia nazwy zbioru pierwotnego (spacja, a po niej nazwa TAIL);
- wybrania opcji Run (klawisz Alt-R).
Wykonanie programu dwumodułowego zawartego w zbiorach FIRST.C i SECOND.C (wydruk 4) wymaga:
- utworzenia zbioru name.PRJ (np. PROGRAM.PRJ) zawierającego opis projektu; w najprostszym przypadku będzie to wykaz nazw jego zbiorów składowych (FIRST SECOND);
- zgłoszenia systemowi Turbo C nazwy projektu przez wybranie opcji Project (klawisz Alt-P), a następnie wybranie podopcji Project name (klawisz P);
- określenia nazwy projektu (spacja, PROGRAM);
- skompilowania i wykonania programu (klawisz Alt-R).



1. Wybór kodów okna edycji
2. Możliwości optymalizacyjne kompilatora Turbo C

Turbo C/Turbo Pascal

Sterowanie kursora

Ctrl S kursor w lewo
Ctrl D kursor w prawo
Ctrl E kursor w górę
Ctrl X kursor w dół
Ctrl A poprzedni wyraz
Ctrl F następny wyraz
Ctrl W przewinięcie w górę
Ctrl Z przewinięcie w dół
Ctrl R strona w górę
Ctrl C strona w dół
Ctrl Q-S na początek linii
Ctrl Q-D na koniec linii
Ctrl Q-E na górę ekranu
Ctrl Q-X na dół ekranu
Ctrl Q-R na początek tekstu
Ctrl Q-C na koniec tekstu
Ctrl Q-B na początek bloku
Ctrl Q-K na koniec bloku
Ctrl Q-P poprzednia pozycja kursora

Wstawianie i kasowanie tekstu

Ctrl V tryb wstawiania/nadpisywania
(DEL) kasuj znak z lewej
Ctrl G kasuj znak pod kursorem
Ctrl T kasuj wyraz z prawej
Ctrl N wstaw linię
Ctrl Y usuń linię
Ctrl Q-Y kasuj wszystkie znaki do końca linii

Operacje blokowe

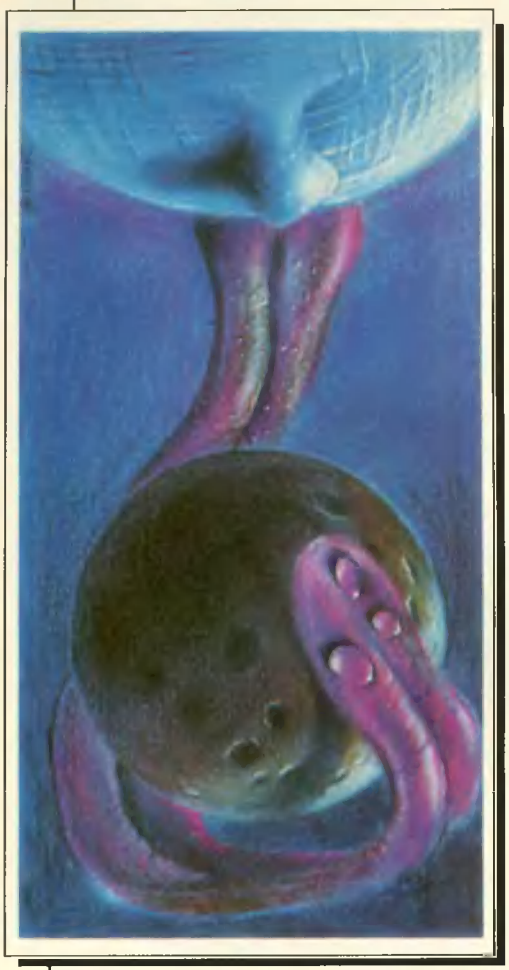
Ctrl K-B zaznacz początek bloku
Ctrl K-K zaznacz koniec bloku
Ctrl K-T zaznacz pojedynczy wyraz
Ctrl K-H ukryj/wyświetl blok
Ctrl K-C skopiuj blok
Ctrl K-V przesuń blok
Ctrl K-Y skasuj blok
Ctrl K-R odczytaj blok z dysku
Ctrl K-W zapisz blok na dysku

Inne funkcje

Ctrl K-D koniec edycji
Ctrl Q-I automatyczne formatowanie
TAB/Ctrl I tabulacja
Ctrl Q-I odtwórz linię
Ctrl Q-F wyszukaj tekst (NIE DZIAŁA!)
Ctrl Q-A wyszukaj i zamień tekst
Ctrl L powtórz wyszukiwanie
Ctrl P prefiks znaków sterujących
Ctrl U przerwanie edycji

Klawisze funkcyjne w Turbo C

F1 przywołanie na ekran informacji pomocniczych (Help)
F2 zapisanie pliku z Edytora na dysku
F3 inicjacja ładowania zbioru
F5 rozszerzenie bieżącego okna do rozmiaru pełnego ekranu lub przywrócenie pierwotnego rozmiaru
F6 zmiana aktywnego okna
F7 ujawnienie poprzedniej usterki
F8 ujawnienie następnej usterki
F9 kompilacja i konsolidacja programu
F10 wywołanie głównego menu
Alt F1 wyświetlenie ostatnio przywołanych informacji pomocniczych
Alt F3 inicjacja wyboru zbioru do załadowania
Alt F9 kompilacja pliku znajdującego się w Edytorze
Alt X powrót do systemu MS DOS



Wielokrotne wykonanie programu wymaga:
- utworzenia programu wykonywalnego (klawisz F9);
- chwilowego opuszczenia systemu Turbo C i wywołania systemu MS-DOS (klawisz Alt-F, a następnie klawisz O);
- wielokrotnego wykonywania programu w systemie MS-DOS;
- powrotu do programu Turbo C (dyrektywa Exit).

Uruchamianie programów

Podstawowym warunkiem uruchomienia programu jest usunięcie zawartych w nim błędów składniowych. W tym celu, bezpośrednio po przygotowaniu programu źródłowego, należy poddać go kompilacji, a następnie przeanalizować wszystkie wykryte przez kompilator usterki programowe. Kompilowanie modułu źródłowego, znajdującego się w buforze edytora, jest inicjowane klawiszem Alt-F9. Jeśli kompilacja przebiegnie pomyślnie, pojawi się komunikat: Success: Press any key. Jeśli natomiast podczas kompilacji wykryte zostaną błędy, jest to zgłoszone komunikatem: Errors: Press any key.
W celu poprawienia wykrytej usterki należy wybrać odpowiedni komunikat błędu, posługując się klawiszami kursora (góra, dół) i nacisnąć klawisz Enter. Oknem aktywnym stanie się okno edycji, a kursor znajdzie się w pobliżu tej usterki, która spowodowała wyprowadzenie komunikatu. Po wprowadzeniu poprawek można przywrócić aktywność okna komunikatów klawiszem F6. Można także, pozostając nadal w oknie edycji, posłużyć się klawiszami funkcyjnymi F8 (F7), w celu zlokalizowania kolejnej (poprzedniej) usterki. Po usunięciu wszystkich błędów można powtórzyć kompilację albo spowodować kompilację i wykonanie programu (klawisz Alt-R). Zanim zostanie podjęta taka decyzja, można dowolnie często przełączać okna edycji i komunikatów (klawiszem F6).

Proces identyfikowania usterek jest znacznie ułatwiony dzięki temu, iż przenoszeniu się z wiersza do wiersza w oknie komunikatów towarzyszy automatyczne przemieszczanie się wyróżnionego wiersza w oknie edycji. Pewien problem powstaje jedynie wtedy, gdy w oknie komunikatów nastąpi przejście do opisu usterki znajdującej się w innym zbiorze niż ten, który aktualnie znajduje się w edytorze. Ustaje wówczas przemieszczanie się wyróżnionego wiersza w oknie edycji; załadowanie zbioru zawierającego usterkę następuje dopiero po wybraniu komunikatu o usterce (klawisz Enter). Ponieważ w niektórych sytuacjach może być to niewygodne, zapewniono środki umożliwiające automatyczne ładowanie do okna edycji potrzebnych zbiorów. Uzyskanie takiego efektu wymaga jednak zmiany jednej z opcji kompilatora. W tym celu, najlepiej z głównego menu i przed podjęciem kompilacji programu, należy wybrać opcję Debug (klawisz D), następnie opcję Track messages (klawisz T) i tyle razy nacisnąć klawisz Enter, aż pojawi się napis All files (wszystkie zbiory).
Skompilowanie programu z wydruku 5, w którym przed słowem HELLO omyłkowo opuszczono cudzysłów, spowoduje zasygnalizowanie - w oknie komunikatów - następujących opisów usterek:
[Undefined symbol 'Hello' in function main
[Undefined symbol 'I' in function main
[Function call missing) in function main
Jeśli w tym momencie wybrany zostanie opis usterki (klawisz Enter), to nastąpi uaktywnienie okna edycji i zlokalizowanie pierwszego błędu. Lokalizacja pozostałych błędów może nastąpić za pomocą klawiszy F6, F7 i F8.

Wydruk 1 Procedura wsadowa TURBO-C.BAT

```
MD JB-TC
CD JB-TC
PAUSE DYSKIETKA NR 1 DO A:
COPY A:TC.EXE
COPY A:TCHELP.TCH
COPY A:TCCONFIG.TC
MD INCLUDE
CD INCLUDE
PAUSE DYSKIETKA NR 3 DO A:
COPY A:*.H
MD SYS
CD SYS
COPY A: \SYS \STAT.H
CD .. \..
MD LIB
CD LIB
COPY A:CO?.OBJ
COPY A:EMU.LIB
COPY A:MATH?.LIB
COPY A:C?.LIB
CD .. \..
PAUSE KONIEC
```

Wydruk 2

```
#include <stdio.h>
main()
{
   printf("Hello, I am Jan B");
}
```

Wydruk 3

TAIL.C

```
#include <stdio.h>
main()
{
#include HEAD.H"
}
```

HEAD.H

```
   printf("Hello!");
```

Wydruk 4

FIRST.C

```
#include <stdio.h>
main()
{
   display();
}
```

SECOND.C

```
   display()
   {
      printf("That's all");
   }
```

Wydruk 5

```
#include <stdio.h>
main()
{
   printf("Hello, I am Jan B");
}
```


Oferujemy do sprzedaży:

- mikrokomputery IBM PC XT/AT, Personal System/2, oraz 32–bitowe
- mikrokomputery AMSTRAD–SCHNAIDER
- urządzenia peryferyjne: drukarki, stacje dysków 3" i 5,25", dyski twarde, monitory, terminale, plotery, streamery i in.
- oprogramowanie użytkowe
- magnetowidy, kamery, monitory, kasety magnetowidowe.

Udzielamy gwarancji, zapewniamy serwis pogwarancyjny i materiały eksploatacyjne.

Spółdzielnia Pracy UNICUM
Dział Komputerów
00-504 Warszawa 15
skr. pocz. 20
Tel.49-56-66

unicum

Czytelnicy i komputery

W lipcowym numerze pisma Byte (1987 r.) zamieszczono ciekawe informacje o użytkownikach mikrokomputerów, wykorzystywanym przez nich sprzęcie i oprogramowaniu. Zródłem informacji jest ankieta przeprowadzona wśród prenumeratorów miesięcznika.

Połowa prenumeratorów liczy mniej niż 36,5 lat, 89% zajmuje stanowiska kierownicze lub prowadzi prywatną praktykę zawodową, a średni dochód roczny prenumeratora wynosi 49,4 tys. dol. Na lekturę miesięcznika (wraz z reklamami ok. 360 stron) czytelnicy poświęcają przeciętnie 3 godziny.

Ankieta potwierdza dominującą w USA pozycję komputerów IBM. W pracy 92,7% czytelników używa lub nadzoruje mikrokomputery IBM (lub z nimi zgodne), a tylko 19,2% - komputery firmy Apple, głównie typu Macintosh. Blisko 83% czytelników stosuje dyski sztywne, 28% - drukarki laserowe (27% planuje zakup takiej drukarki), 41% posiada mysz, a ponad 17% - wyposażenie do sieci lokalnych (16% planuje zakup takiego wyposażenia).

Najpopularniejszym - dla ponad 93% ankietowanych - zastosowaniem mikrokomputerów jest redagowanie tekstów. Arkusze obliczeniowe i programy obsługi baz danych wykorzystuje po 72% respondentów, 58% - programy komunikacyjne, 51% - programy graficzne; własne opracowuje 49% ankietowanych.

Ponad 84% ankietowanych ma mikrokomputer także w domu, a 50% zamierza go w tym roku kupić. Wykorzystanie komputera w domu związane jest przede wszystkim z pracą zawodową (co stwierdziło 72% respondentów), a kontakt z biurem zapewniają modemy, które posiada już ponad 60%, a zamierza kupić dalsze 18% ankietowanych. W domu użytkownicy mikrokomputerów stosują programy analogiczne jak w pracy, a dodatkowo wykorzystują programy do planowania finansowego, edukacyjne i rekreacyjne. Wyposażenie komputerów w domu jest skromniejsze niż w pracy, o czym świadczy fakt, że tylko 47% systemów jest wyposażone w dyski sztywne.

Ankietowani czytelnicy dzielą się chętnie swoją wiedzą o komputerach i oprogramowaniu, doradzając w sprawie zakupu przeciętnie 6 razy w miesiącu oraz udzielając blisko 14 porad miesięcznie na temat zastosowań.

Byte **(rs)**

Olivetti 386 i ...

Plany firmy Olivetti dotyczące wprowadzenia do produkcji nowego komputera bazującego na procesorze 80386 nie były zaskoczeniem dla obserwatorów rynku. Stało się nim natomiast przedstawione przez tę firmę oprogramowanie zwane Topjob. Program Topjob zajmuje ok. 40 KB pamięci, pozwala podzielić pamięć RAM komputera na kilka części, w których można jednocześnie i niezależnie uruchomić kilka różnych zadań, nawet pracujących pod kontrolą różnych systemów operacyjnych. Jest to możliwe dzięki zmuszeniu procesora do pracy imitującej działanie sieci.

Pierwsze przykłady zademonstrowane przez przedstawicieli Olivetti polegały na jednoczesnym uruchomieniu zadań MS-DOS i PC-DOS, realizujących programy Sublogic Jet i Microsoft Flight Simulator (który zwykle uznawany jest za program nie nadający się do pracy w środowisku wielozadaniowym). Bezkolizyjna praca tych programów pozwala przypuszczać, że Topjob może wkrótce stać się przebojem rynkowym.

Your Computer **(JaW)**

Ryszard Ciarkowski

Mowa jest najstarszym i najbardziej naturalnym środkiem porozumiewania się ludzi. Do niedawna była ona wyłącznym atrybutem Homo sapiens. Nowoczesne maszyny wymagają coraz efektywniejszych sposobów komunikacji z człowiekiem, który wydaje im polecenia i kontroluje ich działanie. Komunikacja ta musi być przede wszystkim skuteczna, ale też mało absorbująca, łatwa do opanowania i naturalna. Stąd też z zagadnieniem komunikacji człowieka z maszyną wiążą się badania nad językami naturalnymi, sztuczną inteligencją, metodami analizy i syntezy głosu ludzkiego.

Mowa syntetyczna

Mówiące maszyny kojarzą się bardziej z nowoczesnymi zabawkami niż z „monitorami akustycznymi" - elementami systemów elektronicznych, pozwalającymi odciążyć wzrok użytkownika. Elektroniczna synteza mowy nie jest obecnie rozwiązaniem powszechnym. Tymczasem obecny poziom techniki umożliwia już stosowanie układów syntezy mowy (nie tylko zrozumiałej, lecz nawet o naturalnym brzmieniu) w systemach sterowania, aparaturze kontrolnej i pomiarowej, a także w wysokiej klasy sprzęcie powszechnego użytku. Pozwala to przypuszczać, że niedługo syntetyzatory mowy będą wykorzystywane jako urządzenia wyjściowe równie często, jak dzisiaj monitor ekranowy.

Zastępowanie lub wspomaganie wskazań optycznych komunikatami słownymi jest szczególnie pożądane tam, gdzie uwaga operatora musi być podzielona między wiele wskaźników (np. w systemach automatyzacji procesów produkcyjnych) lub tam, gdzie ingerencja użytkownika wymagana jest jedynie sporadycznie (np. w momencie wystąpienia zakłóceń). Mowa syntetyczna jest też stosowana w łączności radiowej i telefonicznej:

- przy meldunkach stacji bezzałogowych,
- w komputerowej informacji telefonicznej,
- przy tworzeniu systemów transmisyjnych o małym paśmie przenoszenia.

Wraz z rozwojem techniki następowała ewolucja metod zapisu sygnału mowy (rys. 1). Zapis sygnału mowy metodą 1 następuje bezpośrednio ze wzmacniacza na taśmę magnetofonową (analogowo). W metodzie 2 sygnał jest kodowany przez przetwornik analogowo-cyfrowy (A/C) i wpisywany do pamięci cyfrowej. Metoda ta wymaga jednak bardzo dużej pamięci. Metody 3 i 4 wymagają jej już znacznie mniej, ponieważ eliminuje się w nich informację „nadmiarową".

Metody syntezy mowy można też klasyfikować ze względu na wymaganą szybkość transmisji kodów (rys. 2). Trzy zasadnicze grupy metod różnią się jakością „produkowanej" mowy, stopniem kompresji danych i wielkością wymaganej pamięci. Tradycyjna metoda bezpośredniego zapisu sygnału mowy w postaci cyfrowej wymaga szybkości transmisji rzędu 100 Kbitów/s. Synteza mowy o małym stopniu redukcji danych (1:5) wykorzystuje metody ADPCM (Adaptive Differential PCM - adaptacyjna różnicowa modulacja kodowo-impulsowa) i CVSD (Continuously Variable Slope Delta-modulation - adaptacyjna modulacja Delta) i wymaga szybkości transmisji 16 Kbitów/s (rys. 3). Pozwala ona uzyskać mowę syntetyczną o wysokiej zrozumiałości, naturalnym brzmieniu i dużej dynamice. Wymaga znacznej pamięci, co jednak - wobec ciągle malejących cen pamięci półprzewodnikowych - nie stanowi istotnego ograniczenia.

Modulację kodowo-impulsową (PCM) można traktować jako trzy odrębne procesy: próbkowania, kwantowania i kodowania. W efekcie próbkowania otrzymuje się ciąg próbek informujących o amplitudzie sygnału analogowego. Porównanie wartości próbek z poziomami odniesienia w procesie kwantowania prowadzi do określenia skwantowanej wartości próbki sygnału. Na jej podstawie przyporządkowuje się próbce określony zapis kodowy.

ADPCM (wariant DPCM) koduje nie wartość amplitudy sygnału, lecz różnicę amplitudy kolejnych próbek. Sygnał mowy trwający 4 s przy częstotliwości próbkowania 8 kHz wymaga 32 Kbitów pamięci.

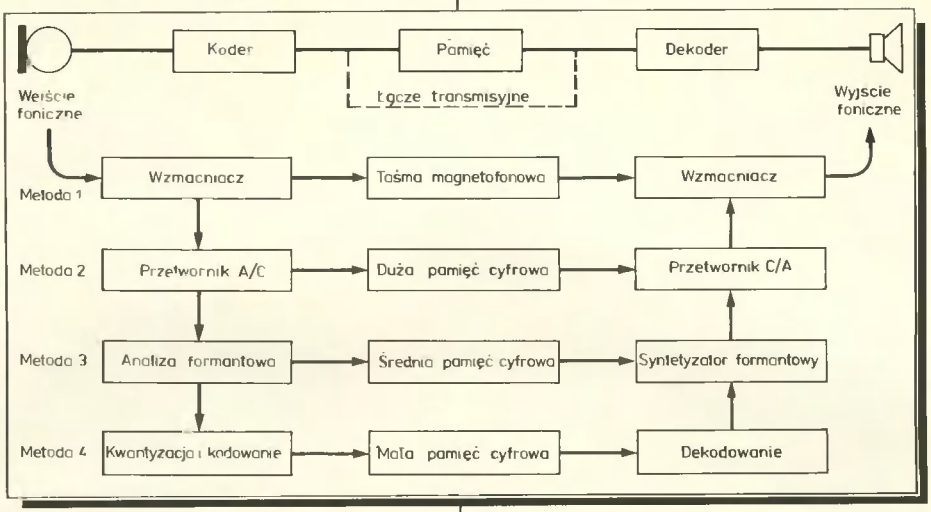
Modulacja Delta również polega na kodowaniu różnicy pomiędzy amplitudami kolejnych próbek, lecz realizuje je w sposób odmienny od modulacji ADPCM. Porównanie próbki sygnału wejściowego z wartością aproksymacji poprzedniej próbki pozwala określić kierunek zmian sygnału wejściowego. Wygenerowana jednobitowa informacja przyjmuje wartość 1 dla przyrostu dodatniego lub 0 - dla przyrostu ujemnego. Wartość kolejnej aproksymacji uzyskuje się przez dodanie (lub odjęcie) do (od) poprzedniej wartości pewnej stałej (tzw. przyrostu). Wadą metody jest możliwość wystąpienia sytuacji, w których następuje pogorszenie jakości sygnału przy jego odtwarzaniu. Jeśli np. sygnał wejściowy gwałtownie rośnie, to sygnał otrzymywany w wyniku przetwarzania nie może za nim nadążyć, gdyż maksymalna szybkość jego zmian odpowiada dokładnie jednemu przyrostowi na jeden okres próbkowania. Błąd powstanie też wtedy, gdy różnica dwóch kolejnych próbek sygnału wejściowego będzie mniejsza od pojedynczego przyrostu. Skutkiem tego błędu są oscylacje otrzymanego po przetworzeniu sygnału wokół faktycznej wartości (tzw. szum granulacji).

W adaptacyjnej modulacji Delta (CVSD) dla wyeliminowania wspomnianych błędów wprowadzono zmienną wartość przyrostu, rosnącą lub malejącą w zależności od szybkości zmian sygnału. W celu poprawienia jakości

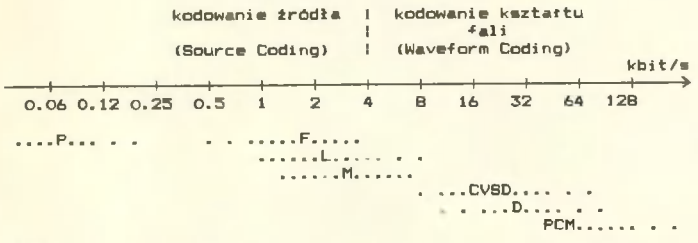
wykorzystuje się również kompandorowanie (kompresja sygnału przed jego zmodulowaniem i ekspansja po demodulacji). Układy stosujące metodę CVSD na ogół obejmują zarówno funkcję kodowania, jak i dekodowania. W dekoderze konieczne jest użycie filtru o tzw. stromej charakterystyce (najczęściej z przełączaną pojemnością) dla stłumienia częstotliwości taktującej i jej harmonicznych. Przy częstotliwości próbkowania 16 kHz zapis czterosekundowego sygnału mowy wymaga 8 Kbitów pamięci.
Uwzględnienie sposobu powstawania dźwięków w torze artykulacyjnym człowieka stworzyło nowe możliwości syntezy mowy z mniejszej liczby danych. Kodowanie nie jest tu realizowane na podstawie analizy zmian sygnału w czasie, lecz polega na wyznaczeniu parametrów kształtujących charakterystykę toru artykulacyjnego. Wymaga to jednak złożonej analizy dźwięku. Przedmiotem analizy nie jest sygnał zmieniający się w czasie,

Rozwiązania układowe

Metody syntezy mowy znane są od dawna, lecz ich realizacja ze względu na duże wymiary i wysokie koszty urządzeń ograniczały możliwości powszechnego stosowania. W latach siedemdziesiątych opracowany został syntetyzator formantowy **COMPUTALKER CT-1** amerykańskiej firmy Computalker Consultants. Jest to pakiet zawierający analogowy układ syntetyzujący, zrealizowany w technice hybrydowej (moduł źródła tonu krtaniowego, źródła szumu i moduł kształtowania widma sygnału) oraz układ sterowania cyfrowego. W podobnej formie oferuje się obecnie wysokiej klasy układy syntetyzujące przeznaczone do zastosowań badawczych.
Rozwój mikroelektroniki pozwolił na zredukowanie rozmiarów układów syntetyzatora do jednej monolitycznej struktury zajmującej po-



1. Ewolucja metod realizacji zapisu sygnału mowy prowadząca do syntezy



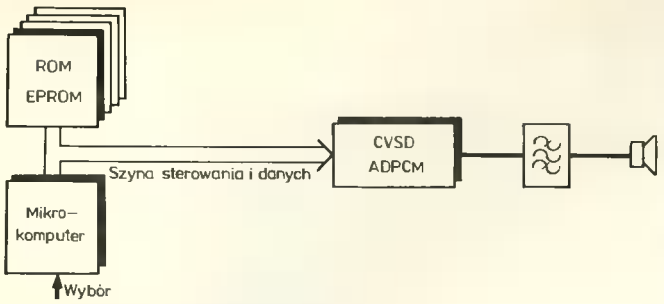
2. Szybkość transmisji danych dla różnych metod kodowania sygnału
P - synteza fonemowa; F - synteza formantowa; L - synteza LPC; M - kodowanie mozerowskie; CVSD - Continuously variable slope delta modulation;
D - modulacja Delta; PCM - Plus-Code Modulation

łecz źródła dźwięków mowy i tor artykulacyjny (ich charakterystyki częstotliwościowe). Syntezy dokonuje się przez elektroniczną symulację toru artykulacyjnego. Metoda ta wykorzystująca model typu „źródło - filtry" (rys. 4) zapewnia znaczną redukcję danych (1:50). Dotyczy ona syntetyzatorów formantowych i LPC (Linear Predictive Coding - kodowanie z predykcją liniową), wykorzystywanych dla małego lub średnio licznego słownictwa, zapisanego w pamięci stałej ROM lub EPROM. Typowa szybkość transmisji danych sterujących wynosi 1,2...3,6 Kbitów/s. Rozwiązanie to jest kompromisem pomiędzy jakością mowy a wielkością pamięci.
Synteza formantowa jest syntezą mowy ze składowych częstotliwościowych, stanowiących maksima energetyczne sygnału mowy (częstotliwości rezonansowe toru artykulacyjnego zwane są formantami).
Synteza LPC także polega na symulacji toru artykulacyjnego człowieka. Współczynniki LPC wyznacza się poprzez zamianę na postać cyfrową i analizę ograniczonego w czasie segmentu mowy.
Do metod syntezy o wysokiej redukcji danych zalicza się także **kodowanie Mozera**. Obejmuje ono próbkowanie i kwantyzację sygnału, podobnie jak ADPCM i modulacja Delta. Na etapie właściwego kodowania uwzględniane są reguły dotyczące kształtowania parametrów częstotliwościowych mowy (dźwięczność/bezdźwięczność, normalizacja przebiegów częstotliwości tonu krtaniowego, kompensacja fazy). Pozwala to na redukcję danych w stosunku do ADPCM i modulacji Delta.
Pełną syntezę głosek - tzw. **syntezę fonemową** - (rys. 5) charakteryzuje mała szybkość transmisji danych (ok. 100 bitów/s), mała pamięć, lecz równocześnie słaba jakość „produkowanej" mowy (tzw. brzmienie komputerowe). W pamięci przechowuje się zapis pojedynczych dźwięków mowy - głosek. Dla języka angielskiego jest ich ok. 65, niemieckiego - 50. Układy sterujące w syntetyzatorze pozwalają łączyć te elementy w słowa według pewnych uproszczonych reguł. Zaletą tej metody jest nieograniczone słownictwo. Możliwe jest także przetwarzanie tekstu pisanego na mowę syntetyczną (tzw. system Text-to-Speech).

Tabela 1. Scalone układy syntezy mowy

Producent	Oznaczenie układu	Metoda syntezy	Pamięć wew. ROM	Technologia
AMI	S 3610	LPC	20 K	CMOS
	S 3620	LPC	16 K	CMOS
General Instr.	SP 0200	formantowa		NMOS
	SP 0232	formantowa	32 K	NMOS
	SP 0250	formantowa		NMOS
	SP 0256	formantowa	16 K	NMOS
	SP 0562	LPC		
	SP 1000	LPC		
	SP 0255 AL2	z głosek		
	VSM 2128 AL2	z głosek		
Harris	HC 55564	CVSD		CMOS
	HC 55516/55532	CVSD		CMOS
Hitachi	HD 38880	LPC		PMOS
	HD 61875/7	LPC	4 K	CMOS
	HD 61885	LPC	32 K	CMOS
ITT	UAA 1003	wg Mozera	28 K	NMOS
	UAA 1103	wg Mozera	28 K	NMOS
	UAA 1104	LPC		HMOS
	UAA 1105	LPC		HMOS
MEC	MN 6401	LPC	32 K	NMOS
Mitsubishi	M 50800	LPC	64 K	
	M 50802	LPC	32 K	
	M 58817	LPC		PMOS
Motorola	MC3417/MC3517	CVSD		
	MC3418/MC3518	CVSD		
NEC	µPD 7751	ADPCM		NMOS
	µPD 7752	formantowa	32 K	CMOS
National Semicond.	MM 54104 (SPC)	wg Mozera		NMOS
OKI	MSM 5205 RS	ADPCM		CMOS
	MSM 5218	ADPCM		CMOS
	MSM 5248	ADPCM	48 K	CMOS
	MSM 6202	ADPCM	144 K	CMOS
	MSM 6212	ADPCM	288 K	CMOS
	MSM 6243	ADPCM	192 K	CMOS
Philips/ Valvo	MEA 8000	formantowa		NMOS
Suwa Seikosha	SVM 9300	PCM		CMOS
Telesensory	CRC	wg Mozera		PMOS
	Speech 1000	LPC	458 K	
	Series III	wg Mozera	128 K	PMOS
Texas Instr.	TMS 5100/5200/ 5220	LPC		PMOS
Votrax/ Silicon Systems Inc.	SC-01	z głosek		CMOS
	SSI 263	z głosek		CMOS

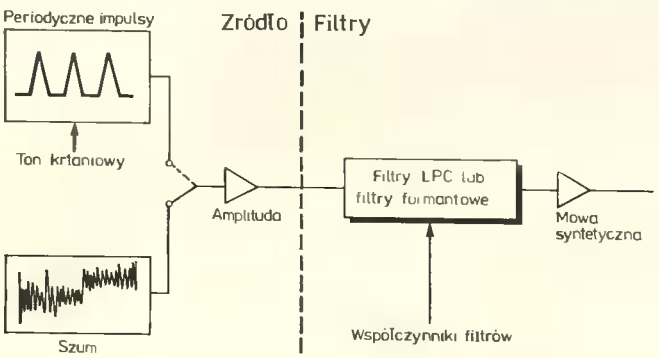
wierzchnię ok. 16...40 mm². Przy takiej samej jak w układzie COMPUTALKER CT-1 strukturze części zasadniczej - szeregowego syntetyzatora formantowego - układ MEA 8000 firmy Valvo (Philips) charakteryzują rozbudowane bloki przyjmowania danych sterujących i wyjścia akustycznego. Dla poprawienia jakości mowy stosowana jest w nich interpolacja liniowa. W stosunku do układu CT-1 pozwoliło to zredukować słowo sterujące syntetyzator z 9 do 4 bajtów, mimo zwiększenia liczby parametrów sterujących syntezę z 9 do 11.



3. Synteza mowy metodą CVSD i ADPCM

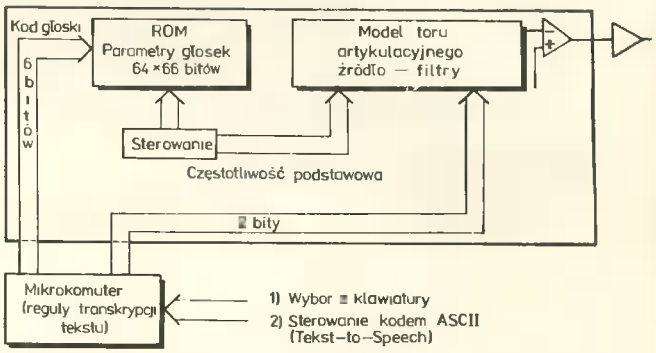
Metodę syntezy ADPCM wykorzystuje rodzina układów MSM serii 52/56 firmy OKI. Układ syntetyzujący **MSM 6218** zawiera w jednej strukturze: pamięć stałą ROM o pojemności 288 Kbitów (można w niej zapisać kody 124 wyrazów), syntetyzator ADPCM z wyjściowym 12-bitowym rejestrem zatrzaskującym oraz 10-bitowy przetwornik A/C, współpracujący z zewnętrznym wzmacniaczem. Wewnętrzna pamięć układu pozwala na syntezę wypowiedzi trwającej do 40 s, przy pracy z szybkością transmisji kodów mowy 7...32 Kbitów/s. Kolejne wersje tej serii różnią się pojemnością wewnętrznej pamięci ROM. OKI oferuje również serię układów zwaną **Real Voice**, złożoną z syntetyzatora/analizatora **MSM 5218**, syntetyzatora **MSM 5205** oraz pamięci EPROM dla kodów ADPCM - **MSM 2764 RS**. Syntezę ADPCM realizuje także układ μPD 7751 firmy NEC, który przy szybkości transmisji parametrów syntezy 14...20 Kbitów/s zapewnia dobrą jakość syntetyzowanej mowy, niezależną od mówiącego (typu głosu, według którego opracowuje się kodowy zapis mowy).
Układy przetwarzające mowę na zasadzie adaptacyjnej modulacji Delta (CVSD) łączą zarówno funkcje koderów, jak i dekoderów. Zalicza się do nich układy MC 3417/3418 firmy Motorola oraz HC 55516/55532 i HC 55564 firmy Harris.

Metodę pośrednią pomiędzy LPC i syntezą formantową - kodowanie Mozera - zastosowano w jednoukładowym syntetyzatorze firmy ITT UAA 1003. Dzięki niskiej cenie znalazł on szerokie zastosowanie w zegarkach (synteza komunikatów o czasie zegarowym), automatach telefonicznych oraz do generowania komunikatów informacyjnych i alarmowych. Obok bloku sterowania dostosowanego do współpracy z mikroprocesorem oraz bloku dekodera i przetwornika C/A zawiera on wewnętrzną pamięć ROM o pojemności 28 Kbitów, mogącą (dzięki redukcji danych) zapamiętać ok. 20 zakodowanych wyrazów. Udoskonalona wersja tego układu (UAA 1103) pozwala zapisać w pamięci o tej samej pojemności kody aż 32 wyrazów. Firma National Semiconductor produkuje pod nazwą „Digitalker" zestaw złożony z generatora mowy MM 54104, pamięci ROM 128 Kbitów oraz zewnętrznych elementów generatora taktu oraz filtru i wzmacniacza. Zestaw współpracuje z większością mikroprocesorów i umożliwia syntezę 256 słów, które - według życzenia użytkownika mogą być wypowiadane głosem żeńskim, męskim lub dziecięcym.
Wspomniany już syntetyzator formantowy MEA 8000 firmy Valvo (Philips) zapewnia możliwość współpracy z większością 8-bitowych mikroprocesorów. Parametry sterujące syntezą przesyłane są 8-bitową szyną danych. Średnia szybkość transmisji parametrów syntezy wynosi ok. 1000 bitów/s. Obsługa syntetyzatora zajmuje jedynie ok. 1% czasu pracy procesora. Inny popularny syntetyzator formantowy to układ μPD 7752 firmy NEC. Wymagana dla niego szybkość transmisji danych wynosi 2,4 lub 5,6 Kbitów/s. Wewnętrzna 32-Kbitowa pamięć ROM pozwala na zakodowanie 63 wyrazów (trwających 13 s przy prędkości transmisji 2,4 Kbita/s). Posiada on zarówno wyjście analogowe poprzez 9-bitowy przetwornik C/A, jak również szeregowe wyjście cyfrowe. Układ jest dostosowany do współpracy z mikroprocesorami 8080A, 8085A i 8084.
Syntezy głosek użyto w układach SC-01 i SSI 263 firmy Votrax i SP 0255-AL2 firmy General Instruments. Dla układu SC-01 opracowano metodę zapisu parametrów głosek, umożliwiającą zmniejszenie szybkości transmisji danych do ok. 70 bitów/s. W wewnętrznej pamięci ROM zapisane są parametry syntezy dla 61 głosek języka angielskiego i trzech pauz pomiędzy nimi. Syntezę realizuje się w oparciu o metodę formantową. Wybór głosek realizowany jest przez mikroprocesor. W trakcie przejścia od jednej głoski do drugiej każdy parametr przechodzi liniowo (w ośmiu krokach) do nowej swej wartości. Wykorzystując ten syntetyza-



4. Model układu źródło-filtry

Syntetyzatorami wykorzystującymi metodę LPC są układy UAA 1104 i UAA 1105 firmy Intermetall/ITT stosowane w telekomunikacji, elektronice przemysłowej i przemyśle motoryzacyjnym. UAA 1104 został przewidziany dla niezależnych systemów syntezy mowy, złożonych z mikroprocesora sterującego, dużej pamięci stałej oraz syntetyzatora LPC. UAA 1105 zawiera rozbudowany interfejs, umożliwiający współpracę z mikrokomputerem. Jakość mowy syntetyzowanej przy użyciu tych układów zależy od szybkości przesyłania parametrów syntezy, zapisanych w pamięci stałej ROM do generatora mowy. Średnią jakość mowy gwarantuje prędkość 3,6 Kbitów/s. O słownictwie decyduje szybkość transmisji oraz pojemność pamięci ROM wraz z możliwościami adresowymi mikroprocesora sterującego. Firma Texas Instruments przedstawiła w roku 1978 pomocną w nauce zabawkę - urządzenie o nazwie „Speak and Spell". Składa się ono z syntetyzatora mowy TMS 5100 z filtrem LPC dziesiątego rzędu, 131-Kbitowej pamięci stałej z zapisem parametrów mowy oraz mikroprocesora. Zawiera klawiaturę z literami alfabetu umożliwiającą napisanie wybranego słowa oraz wyświetlacz, na którym pojawia się syntetyzowane słowo.



5. Synteza mowy z głosek

tor firma Standard Microsystems opracowała kartę syntezy mowy „PC Talker" dla mikrokomputerów IBM-PC. Po zainstalowaniu modułu można za pomocą mikrokomputera syntetyzować dowolny tekst w języku angielskim. Syntetyzator może być programowany w BASICu. Układ SSI 263, przewidziany do współpracy z mikroprocesorami 8-bitowymi, pozwala na rozszerzenie zestawu głosek do 256.
Technologia produkcji układów scalonych wielkiej skali integracji (VLSI) umożliwia połączenie wielu funkcji związanych z mową syntetyczną w jednej strukturze półprzewodnikowej. Takie zadania jak analiza i synteza mowy, tworzenie i przechowywanie słownika oraz konwersja tekstu na mowę będą mogły być realizowane przez pojedynczy układ scalony. Stwarza to perspektywę znacznego upowszechnienia urządzeń głosowej komunikacji człowieka z maszyną.

Ryszard Ciarkowski

Cała μklanowa

Ten numer jest znacznie mniej okazały graficznie niż poprzednie. Przyczyn niezamierzonej skromności jest kilka. Główna – to konieczność pomieszczenia na tych samych 36 stronicach większej liczby artykułów, z racji skrócenia naszego roku wydawniczego, kończonego tym (podwójnie datowanym), a nie – jak planowaliśmy – następnym wydaniem. Dziś w μklanowej galerii prezentujemy tryptyk Marka Zalejskiego (s. 2 i 5) towarzyszący artykułom o języku C oraz – związane z tym samym tematem – okiadkę i ilustrację do spisu treści (s. 1) Zbigniewa Furmana. Obaj twórcy znani są Czytelnikom zapewne nie tylko z mikroklanowych łamów. (red.)

Romuald Szuniewicz

1-2-3

Programy do obliczeń tablicowych, nazywane też elektronicznymi arkuszami obliczeniowymi lub rachunkowymi (ang. electronic spreadsheets), stanowią jedną z najszerzej stosowanych grup programów mikrokomputerowych. Pierwszy z nich o nazwie Visicalc (wprowadzony w 1979 r.) przyczynił się do rozwoju masowej sprzedaży komputerów osobistych.
Prezentując możliwości takich programów za wzorzec przyjęliśmy Lotus 1-2-3, najpopularniejszy program dla PC XT/AT. Obok niego do popularnych elektronicznych arkuszy obliczeniowych należy zaliczyć jeszcze SuperCalc (Computer Associates) i Multiplan (Microsoft).

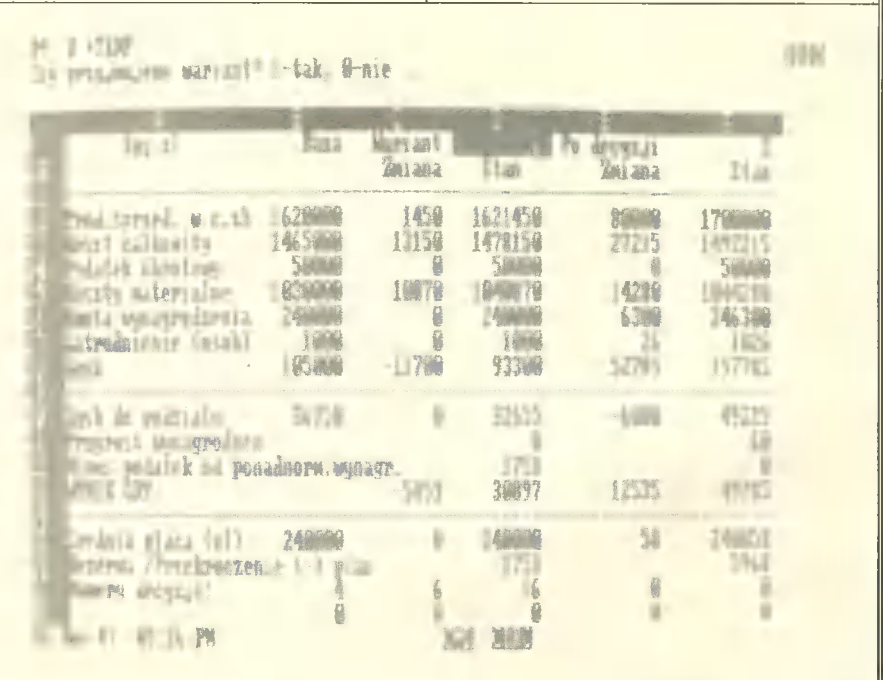
Programy do obliczeń tablicowych pozwalają na wprowadzenie danych liczbowych i tekstu z klawiatury na ekran reprezentujący fragment tablicy, nazywanej arkuszem roboczym (ang. worksheet). Wartości w poszczególnych polach tablicy mogą też być wyliczane na podstawie wielkości zapisanych na innych wskazanych polach. Przy definiowaniu arkusza zamiast liczby wpisywany jest wtedy symbol działania lub zależność funkcyjna.
Oprócz podstawowych funkcji matematycznych, takich jak logarytm i jego odwrotność, pierwiastek kwadratowy, funkcje trygonometryczne, wartość modulo, część całkowita, generator liczb pseudolosowych, w arkuszu można stosować funkcje statystyczne, logiczne oraz tzw. finansowe. Statystyczne obejmują obliczanie sumy, wartości średniej, wariancji i odchylenia standardowego, znajdowanie liczby argumentów, wartości minimalnej i maksymalnej dla zadanego zestawu pól. Funkcje logiczne -działania na wartościach logicznych typu prawda i fałsz oraz operacje warunkowe, a finansowe to operacje matematyczne związane z procentami składanymi. Istnieją także funkcje umożliwiające operacje na datach.
Stosunkowo proste jest definiowanie zależności rekurencyjnych. Możliwe jest też prowadzenie obliczeń iteracyjnych (1...50 iteracji) i automatyczne wyszukiwanie rozwiązań optymalnych.
Nawet do utworzenia złożonego modelu matematycznego nie jest potrzebna znajomość żadnego języka programowania. Wszelkie operacje są wykonywane przez wybór odpowiedniej funkcji z menu. W razie potrzeby można przywołać tekst objaśniający (ang. help). Funkcje wywoływane są przez wpisanie w pole tablicy odpowiedniej nazwy i wskazanie (np. przez podświetlenie) zakresu argumentów (czyli pól tablicy).
Dowolne szeregi liczb (elementów tablicy) mogą być wykorzystane do tworzenia wykresów wyświetlanych na ekranie. Każda wprowadzona zmiana w danych liczbowych powoduje przeliczenie wielkości zależnych i zmianę przebiegu wykresu. Użytkownik może wielokrotnie interakcyjnie lub automatycznie badać odpowiedź modelu na zmianę danych i korygować jego postać. Pozwala to na przeprowadzanie analiz „co by było gdyby," szczególnie przydatnych w ekonomice i planowaniu.
Dowolny fragment tablicy lub wykres mogą być wydrukowane. Duże tablice są drukowane fragmentami na poszczególnych kartkach z marginesem umożliwiającym sklejenie z nich jednego arkusza.

Budowanie arkuszy

W programie 1-2-3 (wersja 1 i 1A) arkusz może mieć wymiary 256 kolumn na 2048 wierszy. Kolumny oznaczane są symbolami literowymi, a wiersze numerami. Na ekranie wyświelanych jest 20 wierszy (przy użyciu monitora monochromatycznego lub EGA 33 wiersze), a liczba widzianych kolumn zależy, oczywiście, od ustalonej przez użytkownika ich szerokości (1...72 znaków). Nazywany komórką element arkusza na przecięciu kolumny i wiersza może zawierać dane liczbowe, tekstowe (etykiety) lub wzory. Każda komórka może pomieścić do 240 znaków alfanumerycznych. Jeśli napisy wprowadzone do komorki przekraczają przyjętą szerokość kolumny, wówczas są wyświetlane na sąsiadujących komórkach, o ile nie są one wypełnione. W programie 1-2-3 nie przewidziano możliwości zmiany wysokości komórki. Liczby są zapamiętywane z dokładnością do 15 cyfr znaczących, a wyświetlane w formacie ustalonym przez użytkownika.
Wyboru komórki, do której mają być wpisane dane, dokonuje się za pomocą klawiszy sterujących kursor (tutaj obrazowany jest on przez podświetlenie). Przeglądanie arkusza odbywa się przez przesuwanie podświetlenia do odpowiedniej komórki lub podanie rozkazu z adresem komórki (np. D23). Możliwy jest też wybór pierwszej nie wypełnionej (albo pierwszej wypełnionej) komórki w wybranym kierunku. Podzielenie ekranu na okna umożliwia oglądanie dwóch różnych fragmentów arkusza równocześnie.
Na górze ekranu w wierszu informacyjnym jest wyświetlana niesformatowana zawartość aktualnie podświetlonej komórki, tzn. liczba, etykieta wraz z symbolem określającym jej położenie w komórce lub formuła. Zawartość komórek można dowolnie zmieniać (istnieje jednak możliwość zabezpieczania wybranych przed zmianą). Każdorazowa zmiana wartości komórki powoduje automatyczne (lub tylko na żądanie użytkownika) odpowiednie zmiany we wszystkich komórkach z nią powiązanych. Istnieje możliwość zbudowania jedno- lub dwuwymiarowej tablicy wrażliwości, w której przedstawiany jest wybrany wynik obliczeń dla wielu założonych wartości jednej lub dwóch wielkości wejściowych, a więc wynik wielokrotnego przeliczania arkusza.
Dowolny prostokątny zestaw komórek (w tym także jedna komórka) może być zdefiniowany jako blok przez wskazanie adresów narożnych. przekątnych komórek bloku lub przez ich podświetlenie. Blok ten można następnie kopiować, kasować, przenosić w inne miejsce arkusza lub zapisać na dysku.
Blokom komórek można nadać nazwy i odwoływać się do nich w operacjach i funkcjach (wzorach). Np. komórka o nazwie „wartość" może być wyliczona z zależności „cena*ilość", gdzie „cena" i „ilość" są nazwami komórek. Można zdefiniować format wyświetlania liczb lub funkcję dla jednej komórki i skopiować ją do dowolnych innych bloków, co wielokrotnie przyspiesza budowę arkuszy
Przy kopiowaniu komórek i bloków zawierających funkcje odwołujące się do innych komórek następuje automatyczne przeadresowanie argumentów względem nowego położenia. Oznacza to, że w nowym miejscu funkcja odwołuje się do komórek o analogicznym położeniu względem komórki, w której jest zapisana. Możliwe jest też stosowanie adresów bezwzględnych, które nie ulegają zmianie przy kopiowaniu.

Operacje na bazach danych

Zapisana w arkuszu tablica może stanowić bazę danych (wiersze tablicy traktowane są jako rekordy, a komórki w kolejnych kolumnach jako pola).



Przy użyciu rozkazów programu można uporządkować całą lub część danych według wartości lub kolejności alfabetycznej (dla wybranej kolumny). Po zdefiniowaniu zestawu kryteriów istnieje możliwość wyszukiwania i wyodrębniania rekordów spełniających wszystkie kryteria. Po zdefiniowaniu przedziałów wartości, można dla wybranego zakresu komórek automatycznie obliczyć rozkład, czyli liczbę wartości mieszczących się w poszczególnych przedziałach. Przykładowo, jeśli tablica przedstawia listę płac, wówczas rekordami są umieszczone w kolejnych wierszach zapisy dotyczące poszczególnych osób, a polami - informacje o poszczególnych

elementach wynagrodzenia. Rekordy można uporządkować według nazwisk lub wynagrodzenia. Na podstawie sformułowanych kryteriów można wyszukać np. wszystkie osoby o określonym stażu pracy zarabiające mniej niż określoną wartość lub sporządzić zestawienie informujące o liczbie osób w poszczególnych przedziałach wynagrodzenia. Jedne pola mogą być wyliczane na podstawie innych. Na przykład dodatek za wysługę lat lub premia mogą być wyliczane według kolumn z wynagrodzeniem podstawowym oraz kolumn z liczbą przepracowanych lat i procentem premii.

Wykresy

Program 1-2-3 umożliwia przedstawienie szeregów liczbowych w postaci wykresów liniowych, słupkowych, słupkowych składanych i kołowych (do 7 szeregów liczbowych na jednym wykresie). Ponadto, na wykresach typu XY można przedstawić szeregi liczbowe w funkcji innego szeregu liczbowego reprezentowanego na osi poziomej. Specyficzne wykorzystanie wykresów typu XY umożliwia tworzenie rysunków, na przykład schematów technologicznych lub obwodów elektronicznych.
Po obliczeniu rozkładu wartości w przedziałach (szeregów rozdzielczych, w których poszczególnym przedziałom przyporządkowane są liczebności lub częstości) można wykreślić krzywe liczebności i histogramy. Dla jednego arkusza można zdefiniować wiele różnych wykresów i nadać im różne nazwy.
Wykresy mogą być skalowane przez użytkownika lub automatycznie. Mogą zawierać dwa wiersze tytułów, opis osi pionowej i poziomej, legendę do poszczególnych szeregów liczbowych, a każdemu punktowi wykresu można nadać oddzielną etykietę. W przypadku pracy z dwoma monitorami, na jednym może być wyświetlana zawartość arkusza, a na drugim wykres.
Jakość wykresów na ekranie zależy od rozdzielczości monitora. Aby zapewnić dobrą jakość wydruków wykresy tworzone w programie 1-2-3 są przekształcane zgodnie z rozdzielczością drukarki uzupełniającym programem o nazwie PrintGraph, umożliwiającym zmianę skali, pozycji i obroty rysunku. Wydruki tego programu należą do najlepszych jakościowo wykresów, jakie uzyskuje się na drukarkach graficznych. Drukowanie wykresu jest jednak bardzo czasochłonne.

Modele i programy

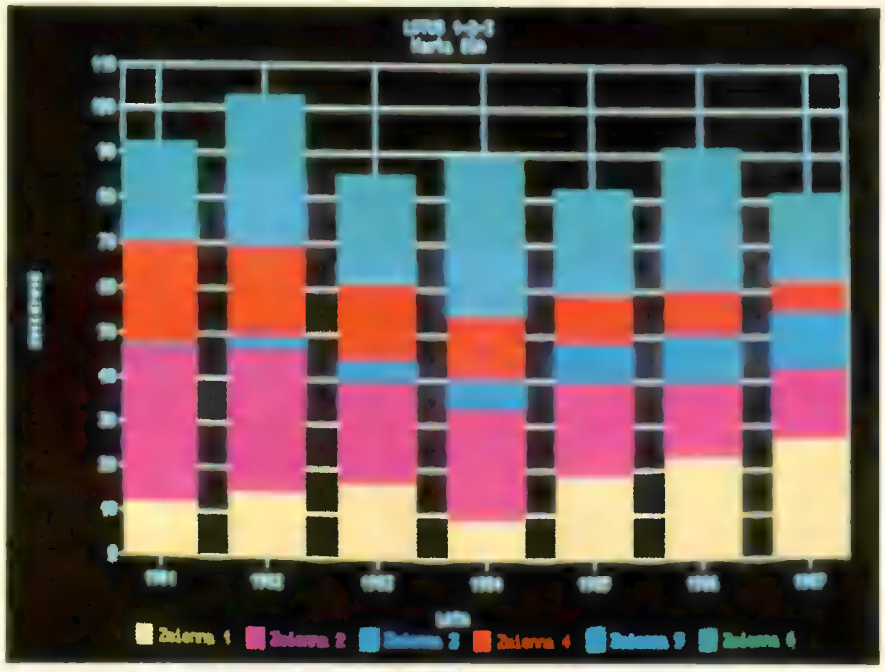
Zapisane w arkuszu dane i zależności mogą reprezentować bazy danych, zjawiska i procesy ekonomiczne, techniczne lub inne związane z różnorodnymi obszarami zastosowań. Każdą taką reprezentację, w której występują powiązania pomiędzy danymi, można nazwać modelem lub szablonem (ang. template). Ten sam model może być wykorzystany z różnymi zestawami danych, co pozwala stosować go przez wielu użytkowników.
Obsługa modelu wymaga często wykonania wielu powtarzalnych operacji, związanych na przykład z wymianą informacji z arkuszami na dysku, dostosowaniem modelu do różnych wariantów danych i wprowadzeniem ich. Ciąg rozkazów wymaganych do przeprowadzenia tych operacji może być zapisany w arkuszu jako tzw. makrorozkaz (ang. macro) i uruchomiony wybraną kombinacją ‹Alt› z klawiszem znakowym. Makrorozkaz może składać się z rozkazów wywoływanych z klawiatury oraz specjalnych rozkazów, umożliwiających wykonanie złożonych zadań. Wybrany makrorozkaz może być wywoływany automatycznie przy wczytywaniu arkusza do pamięci komputera. Praktycznie, makrorozkazy stanowią programy i procedury napisane w specyficznym języku 1-2-3.
Ponieważ makrorozkazy są zapisywane w arkuszu tak, jak pozostałe dane, istnieje możliwość zmiany poszczególnych rozkazów w trakcie wykonywania makrorozkazu przez wpisanie nowego tekstu do odpowiedniej komórki arkusza. Treść procedur zapisanych w postaci makrorozkazów może się więc dynamicznie zmieniać w trakcie ich wykonywania, co stwarza interesujące możliwości dla programisty.
Wielkość pamięci zajmowanej przez arkusz zależy od jego wymiarów i zawartości komórek (program 1-2-3 z nie wypełnionym arkuszem zajmuje 98.116 bajtów). Napisy i wzory zajmują więcej pamięci niż liczby. Tablica o wymiarach 100x100 wypełniona liczbami od 1 do 10.000 zajmuje 39.996 bajtów, a wypełniona 9-znakowymi napisami 149.996 bajtów. Doświadczalnie można sprawdzić, że liczby całkowite zapamiętywane są w arkuszu 1-2-3 na 4 bajtach, a zmiennopozycyjne na 12 bajtach, tak więc tablica 100x100 wypełniona liczbami zmiennopozycyjnymi zajmie powyżej 120 KB.
Modele wymagające pamięci większej niż 640 KB mogą być podzielone na odrębne arkusze przechowywane na dysku, a następnie przetwarzane i konsolidowane zgodnie z programem zapisanym w postaci makrorozkazów.

Wymiana informacji między programami

Dane zapisywane w arkuszach różnych programów do obliczeń tablicowych mają różną strukturę i nie mogą być bezpośrednio przenoszone. Istnieją jednak programy konwersji, które umożliwiają przekształcanie struktur danych tak, aby arkusze opracowane w jednym programie mogły być wykorzystane w innych. Konwersja struktury arkusza związana jest jednak z usunięciem tych informacji, które są specyficzne dla programu pierwotnego, a nie mają odpowiedników w programie docelowym. Struktury danych w programach 1-2-3 mają przedłużenie nazwy .WKS (w wersjach 1 i 1A) i .WK1 w wersji 2. Popularną, uniwersalną strukturą, do której można przekształcić dane dla większości programów jest DIF (Data Interchange Format). Struktury danych typu .DIF i .WKS stają się standardami, tzn. wiele programów od niezależnych wytwórców ma dostosowane do nich struktury danych lub przynajmniej przyjmuje takie struktury jako opcje. Przykładem mogą być programy: statystyczny-Statgraphics lub graficzne-Energraphics i Chart. Powstaje też wiele programów uzupełniających (ang. add-ons), które pozwalają na dalszą obróbkę danych w arkuszach. Programy Spreadsheet Auditor (Consumers Software) i The Cambridge Spreadsheet Analyst (Cambridge Software Collaborative) ułatwiają wyszukiwanie błędów w arkuszach. Rezydujący w pamięci program SQZ! (Turner Hall Publishing) wykonuje automatycznie kompresję arkuszy przy zapisywaniu na dysk, zmniejszając ich rozmiary o 80%. Program 1,2,3 Forecast! (Bruce L.Gates) służy do analizy trendów czasowych. What's Best (General Optimization Inc.) pozwala na optymalizację metodą programowania liniowego. XY:Spread (Intek Solutions, Inc.) ułatwia przenoszenie danych między różnymi arkuszami, dodając jak gdyby trzeci wymiar do tablic. LotusHAL (Lotus) pozwala analizować powiązania w arkuszu i rozpoznaje rozkazy formułowane w sposób zbliżony do składni języka naturalnego. 4Word (Turner Hall Publishing) uzupełnia program 1-2-3 o możliwości redagowania tekstów.
Najnowszą tendencją w rozwoju programów obliczeń tablicowych jest tworzenie arkuszy wielowymiarowych i umożliwienie automatycznej wymiany informacji między pamięcią operacyjną a dyskiem. Pozwala to budować arkusze o objętości większej niż pamięć operacyjna komputera. Programami takimi są m. in.: VP-Planner (Paperback Software International), który pozwala budować arkusze wielkości 17 MB; CalcIT (Martin Marietta Data Systems); Boeing CALC (Boeing Computer Services) - arkusze do 32 MB TM/1 (Sinper Corp.); microCUBE (McDonnell Douglas Comunications).

Romuald Szuniewicz

W artykule wykorzystano fragmenty książki M. Kleibera i R.Szuniewicza „Komputer osobisty typu IBM PC" przygotowywanej do druku przez PWN.



mikrokram

Łapacz myśli

Przy mnogości i różnorodności informacji na niewiele przydają się bazy danych, komputerowe kartoteki czy notatniki (np. SideKick). Dogodny może się natomiast okazać program TORNADO NOTES firmy Micro Logic Corp. Stanowi on adaptację idei elektronicznego arkusza do zastosowań tekstowych.

Program może zarządzać maksymalnie 676 zbiorami, z których każdy może zawierać do 500 notatek o dowolnej treści i objętości. Dysponuje on pewnymi funkcjami edytora tekstów (np. automatyczne przenoszenie wyrazów), umożliwia przenoszenie tekstów między notatkami i plikami tekstowymi standartu ASCII i WordStar, a także pozwala łączyć, przenosić i powielać notatki.

Najbardziej użyteczną funkcją TORNADO NOTES jest inteligentne wspomaganie wyszukiwania. Po naciśnięciu klawisza G (ang. Get) i podaniu dowolnego hasła (wyraz lub urywek zdania) – program automatycznie wyszuka i wyświetli wszystkie notatki, w których jest ono zawarte.

TORNADO NOTES rezyduje w pamięci RAM (może też być używany jako nierezydencyjny), zajmuje ok. 57 KB. współpracuje z kartami MDA, CGA, EGA (w trybie tekstowym) i Hercules. INFO WORLD

Krzysztof Rzymkowski

Badania naukowe wymagają skomplikowanej i kosztownej aparatury. W wykorzystywanym sprzęcie bardzo często znajduje się komputer, który steruje aparaturę, gromadzi dane pomiarowe i przetwarza je. Mimo różnic w przeprowadzanych eksperymentach niektóre bloki sprzętowe można wykorzystać wielokrotnie - jeśli zastosowany zostanie system modularny pozwalający na łatwą adaptację sprzętu do bieżących potrzeb.

CAMAC i inne

Z rozwojem aparatury elektronicznej zmieniała się również definicja systemu modularnego, nazywanego początkowo systemem blokowym, gdyż pojęcie modularności odnosiło się przede wszystkim do rozwiązań mechanicznych. Obecnie systemem modularnym nazywany jest system tworzący zmienną strukturę z niezależnych, wymiennych zespołów. Warunkiem technicznym modularności systemu jest standaryzacja elektryczna i mechaniczna jego elementów. Podstawową jednostką systemu modularnego jest zwykle samodzielne urządzenie elektroniczne (zwane modułem lub blokiem), komunikujące się z innymi urządzeniami według określonego protokółu za pomocą sieci połączeń, zwanych magistralą. Jest ono zbudowane ze znormalizowanych układów, nazywanych również modułami, które stanowią podstawowe „cegiełki" skomplikowanych struktur.

Bloki systemów modularnych umieszczone są zazwyczaj, zgodnie z zaleceniami międzynarodowymi, w znormalizowanej obudowie o szerokości 19". Pozostałe wymiary (wysokość, szerokość i głębokość) przyjmowane są stosownie do potrzeb.

Bloki umieszczane są wewnątrz kaset na numerowanych stanowiskach. W starszych systemach jedno lub dwa stanowiska były uprzywilejowane, przeznaczone dla bloku sterującego zestawu. Bloki połączone są z magistralą. Dawniej używana była jedna magistrala, podczas gdy nowoczesne systemy wykorzystują ich często aż pięć.

Zestaw aparatury umieszczony w jednej kasecie może stanowić samodzielny specjalizowany przyrząd lub może być fragmentem większego, wielokasetowego systemu sterowanego komputerem. Wymiana bloku lub kilku bloków zestawu umożliwia modyfikację przyrządu bez konieczności zmiany pozostałych jego fragmentów.

Obok rozpowszechnionych wcześniej systemów NIM, CAMAC (w ZSRR - WEKTOR) pojawiło się wiele nowych propozycji systemów modularnych wykorzystujących najnowsze możliwości technologiczne i, co najważniejsze, dostosowanych do systemów mikroprocesorowych o różnych długościach słowa danych. Do najbardziej rozpowszechnionych należą systemy:
8 i 16-bitowe - G-64, G-128;
16-bitowe - MULTIBUS I, VME, AMS;
32-bitowe - VME (rozszerzona wersja), MULTIBUS II.

W ośrodkach fizyki jądrowej oprócz systemu VME rozpowszechniony jest również specjalizowany (badania naukowe) system FASTBUS.

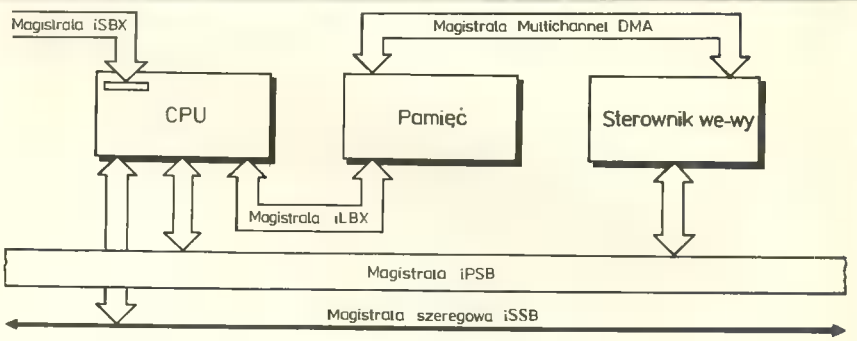
Systemy modularne opracowywane są najczęściej przez koncerny produkujące układy mikroprocesorowe (Motorola - VME, Intel - MULTIBUS). System CAMAC powstał w wyniku współpracy międzynarodowej.

Płytka montażowa typowego bloku ma wymiary eurokarty (Eurocard) 160x112,84 mm lub jej wielokrotności. Najczęściej stosowana jest tzw. podwójna eurokarta 160x233,64 mm. Drugi rozmiar (160 mm), określający głębokość umieszczenia magistrali wewnątrz kasety również może być powiększony. W nowoczesnych systemach przeważnie wykorzystywane są złącza 96-stykowe trójrzędowe lub 64-stykowe dwurzędowe, a nie złącza krawędziowe, jak np. w systemie CAMAC lub MULTIBUS I. Szerokość pojedynczego bloku zależy m. in. od rodzaju użytych złączy. Najczęściej przewiduje się umieszczenie od 20 do 26 stanowisk (przy podwójnej eurokarcie) lub więcej, jeśli w kasecie będą umieszczane bloki o różnych wysokościach. Nowoczesne systemy modularne są na ogół systemami wielomagistralowymi. Zazwyczaj przewiduje się możliwość zamontowania wewnątrz kasety dwóch stałych magistrali wykonanych w postaci sztywnych, wielowarstwowych płyt drukowanych (tzw. plater), stanowiących wraz z łączami i układami dopasowującymi odrębne elementy systemu.

Wszystkie nowoczesne systemy modularne mają zdecentralizowane sterowanie (inteligencję), co wymaga stosowania zupełnie nowych rozwiązań przy tworzeniu oprogramowania, które decyduje coraz częściej o możliwościach systemu i jego cenie. Systemy takie można zatem traktować jako rozbudowane komputery, składające się z różnych elementów spełniających niejednokrotnie rolę autonomicznych bloków sterujących pracę części, jak i - w pewnych okolicznościach - cały system. System jednokasetowy może pełnić funkcję miniaturowej sieci komputerowej.

Protokół komunikacji między blokami systemu musi być ściśle zdefiniowany. W starszych systemach, np. CAMAC, był on dostosowany do pracy hierarchicznej. W kasecie mógł być umieszczony jeden sterownik decydujący o wszystkich przesłaniach przez magistralę systemu. Bloki mogły go jedynie zawiadamiać o wykonaniu powierzonego im zadania, nie mogły natomiast sterować przesłaniami informacji. W nowoczesnych rozwiązaniach przy pracy z wykorzystaniem magistrali głównej systemu, wszystkie bloki są równouprawnione i muszą się ubiegać o dostęp do magistrali. Mechanizmy arbitrażu - ustalania kolejności dostępu - są różnie rozwiązywane. Blok mający dostęp do magistrali staje się jednocześnie sterownikiem całego systemu.

Oprócz magistrali głównej, stosowane są magistrale pomocnicze (tzw. lokalne), o strukturze hierarchicznej, które łączą najczęściej bloki mikrokomputerowe z pamięcią i blokami pomocniczymi. Magistral lokalnych może być w kasecie kilka. Ponadto istnieje na ogół możliwość wykorzystania tzw. magistrali szeregowej o protokole zapewniającym decentralizację inteligencji oraz innych magistrali pomocniczych (np. magistrali we-wy przeznaczonych do bezpośredniej łączności z urządzeniem zewnętrznym). Mikrokomputery systemu mają na ogół dodatkowe indywidualne magistrale do połączeń z urządzeniami peryferyjnymi. W systemie MULTIBUS II przewidywanych jest w sumie pięć niezależnych magistrali.



1. Propozycja nowoczesnego systemu wielomagistralowego firmy INTEL: Multibus II

CAMAC

Jest to system stosowany masowo w różnych dziedzinach nauki i techniki. Wynika to z jego elastyczności i możliwości wykorzystywania najnowszych osiągnięć elektroniki, wprowadzania modyfikacji systemu bez naruszania wymienności funkcjonalnej oraz dobrych rozwiązań funkcjonalnych. Jest to dotychczas jedyny system, który uzyskał międzynarodową aprobatę, nie preferujący wyrobów żadnego konkretnego producenta elementów. Konstrukcja wszystkich bloków systemu, z wyjątkiem bloku pośredniczącego (łączącego zestaw z komputerem lub aparaturą kontrolno-pomiarową), jest niezależna od rodzaju urządzenia sterującego i współpracującej apara-

tury Systemy jednokasetowe mogą być łączone za pomocą magistrali (równoległej lub szeregowej) w systemy wielokasetowe, tworzące rozbudowane sieci automatyczne. Kaseta systemu zawiera 25 stanowisk, z których jedno jest uprzywilejowane, przeznaczone dla sterownika kasety. Magistralą kasety przesyłane są informacje dotyczące adresu urządzenia (numer bloku - stanowiska, adresu wewnętrznego bloku), funkcji, która ma być wykonana, dane oraz zespół sygnałów sterujących i pomocniczych (np. zerowania, taktowania pracy, potwierdzania przyjęcia rozkazu, zgłoszenia, przerwania itp.). Długość słowa danych wynosi 24 bity. Praca systemu jest synchroniczna, czas trwania operacji na magistrali - 1 μs. System pracuje w tzw. logice ujemnej. W wielokasetowym systemie z wykorzystaniem magistrali równoległej może pracować 7 kaset, a z wykorzystaniem magistrali szeregowej 64 kasety. Dodatkowa magistrala ACB umożliwia decentralizację sterowania.

FASTBUS

Bezpośrednią kontynuacją systemu CAMAC i sysemu aparatury analogowej NIM jest system FASTBUS przeznaczony do zastosowań w naukowych ośrodkach fizyki jądrowej. Charakteryzuje się on znaczną szybkością działania. Czas operacji na magistrali wynosi 0,1 μs. Płyta montażowa bloku jest czterokrotnie większa od płyty CAMAC. W kasecie można umieścić 26 bloków spełniających rolę sterowników zapoczątkowujących przesyłanie danych i bloków wykonawczych. Długość słowa adresowego i słowa danych wynosi 32 bity. Adresy i dane są multipleksowane, tzn. przesyłane po tych samych szynach magistrali w różnych fazach cyklu. System może pracować synchronicznie lub asynchronicznie.

G-64 i G-128

System G-64 należy do bardziej rozpowszechnionych ze względu na możliwość stosowania różnych 8-bitowych systemów mikroprocesorowych. Jest on też przystosowany do pracy z mikroprocesorami 16-bitowymi. W systemie przewidziano magistrale 8- lub 16-bitowe do przesyłania danych oraz 16-bitową do przesyłania adresów. System może pracować synchronicznie i asynchronicznie. Jego rozszerzeniem jest system G-128, który ma wiele mechanizmów przejętych z bardziej rozwiniętego systemu VME. Magistrala jest przystosowana do przesyłania danych 8- i 6- oraz 32-bitowych adresów. Również i ten system może pracować synchronicznie i asynchronicznie.

AMS-M

System ten zaproponowany przez firmę Siemens, wykorzystywany jest głównie w ośrodkach przemysłowych. AMS-M zakłada możliwość wykorzystywania dwóch magistrali, tzw. AMS i magistrali lokalnej SMP BUS. Wiele mechanizmów systemu jest zbliżonych do MULTIBUS I. Jest to system przystosowany do pracy ze słowem danych długości 8 i 16 bitów, mający słowo adresowe długości 8, 16 i 32 bity. Czas pojedynczej operacji wynosi 0,3 s. Magistrala AMS systemu jest dostosowana do wyrobów firmy Intel. Praca systemu może być asynchroniczna i synchroniczna. Magistrala AMS jest dostosowana do pracy wieloprocesorowej. Rozwiązanie magistrali lokalnej przewiduje stosowanie jednego bloku sterującego.

NU BUS

Do najnowszych opracowań dostosowanych do wykorzystywania różnych systemów mikroprocesorowych należy system NU BUS. Wiele mechanizmów zostało przejętych z systemu MULTIBUS II. Kaseta systemu może zawierać 16 bloków. Magistralą mogą być przesyłane słowa danych długości 8, 16, 32 bity. Szyny adresów i danych magistrali są multipleksowane. Podobnie jak MULTIBUS II, częstotliwość pracy zegara wynosi 10 MHz, a arbitraż jest zdecentralizowany. Praca systemu jest synchroniczna. Uproszczona wersja magistrali NU BUS została wykorzystana w konstrukcji komputera Macintosh II (mikroklan 6/87).

VME

Jest to najbardziej rozpowszechniony obecnie w Europie system modularny; dostosowany do wyrobów firmy Motorola. Magistrala jest dostosowana do przesyłania słów danych i adresów długości 8, 16 i 32 bity. Szyny adresów i danych nie są multipleksowane. Arbitraż w systemie jest scentralizowany, mimo że system jest przystosowany do pracy wieloprocesorowej. Szybkość pracy systemu jest uzależniona od szybkości pracy systemów mikroprocesorowych używanych do budowy bloków i może być bardzo duża. Magistrala jest przystosowana do pracy synchronicznej. W systemie przewidziano możliwość wykorzystania pięciu różnych magistral pracujących niezależnie. Obecnie wprowadzany jest do systemu mikroprocesor 32-bitowy MC 68020.

MULTIBUS II

Mimo że system ten jest jeszcze w trakcie opracowywania, w firmie Intel rozpoczęto już pierwsze próby budowy zestawów komputerów jednomodułowych (głównie systemy mikroprocesorowe iAPX 256), pamięci oraz bloków pomocniczych do badań prototypowych. Przewiduje się w najbliższych latach ogromny rozwój i rozpowszechnienie systemu. Został on przyjęty przez państwa RWPG jako obowiązujący standard. Rozwiązanie magistrali równoległej systemu jest dostosowane do pracy wieloprocesorowej. Szyny adresów i danych (32-bitowe) są multipleksowane, arbitraż zdecentralizowany. System wykorzystuje pięć niezależnych magistrali, przystosowanych do pracy synchronicznej. Częstotliwość pracy zegara wynosi 10 MHz.

Powszechne zastosowanie wyrafinowanych systemów mikroprocesorowch 16- i 32-bitowy oznacza, że wiele funkcji, które w starszych systemach musiały być realizowane sprzętowo może teraz zostać zrealizowanych w sposób programowy. Zapewnia to większą elastyczność systemu i upraszcza ew. rozbudowę. Jedną z istotnych zalet współczesnych systemów modularnych jest możliwość umieszczania w jednej kasecie bloków o różnych wymiarach, a ściślej - o różnej wysokości, tzn. podwójnej (ostatnio nawet potrójnej) i pojedynczej eurokarty. Pozwala to tworzyć systemy o bardzo elastycznej strukturze przy możliwie najpełniejszym wykorzystaniu miejsca w kasecie, które jest dość kosztowne.

Inną zaletą jest możliwość praktycznie dowolnego skracania magistrali podstawowej systemu. Oferowane są magistrale nawet czterostanowiskowe, umożliwiające budowę niezależnych, samodzielnych komputerów o podstawowej konfiguracji rozszerzonej przez inne urządzenia montowane w tej samej kasecie.

Realizacja wielu współczesnych eksperymentów badawczych wymaga na ogół kilku komputerów ściśle ze sobą współpracujących, rozwiązujących różne zadania. Takie wymagania mogą spełnić współczesne systemy modularne. Zwykle wykorzystuje się ich kilka jednocześnie, głównie NIM i CAMAC powiązanych z G-64 i G-128 oraz FASTBUS, sterowanych komputerami systemu VME współpracującego z dużymi systemami komputerowymi. Ta rozbudowa systemu pomiarowego jest związana z koniecznością zbierania ogromnej liczby danych, co pozwala eksperymentatorom unikać wielokrotnego powtarzania kosztownych doświadczeń.

Przydatne w systemie wyszukiwania informacji tekstowej i pracach projektowo-programistycznych

EMULATOR ZDALNEGO TERMINALU IBM 3270 BSC umożliwiający współpracę IBM PC z systemem IBM 370 lub RIAD

ZETO - ZOWAR
00 - 608 Warszawa
Al.Niepodległości 190
Telex 813492

ADAPTER ADA PT-305 PT-310 umożliwiający współpracę IBM PC z jednostkami pamięci taśmowych typu PT-305.

Roczna gwarancja. * Odpłatny serwis pogwarancyjny. * Szkolenie w zakresie użytkowania

Tel. 25-93-34 Zakład Eksploatacji Systemów Komputerowych

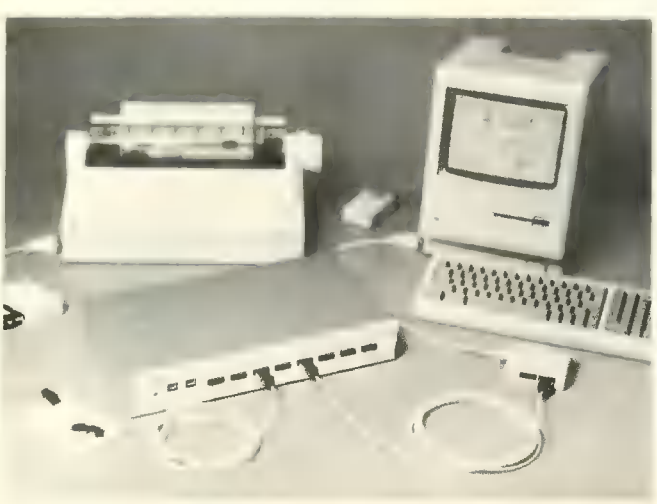
Tel. 25-43-72 Zakład Marketingu i Współpracy z Zagranicą

Szepcząca drukarka

Podczas pracy nowego modelu drukarki - IBM Quietwriter III natężenie hałasu nie przekracza 45 dB, natomiast jakość i szybkość druku dorównuje najlepszym drukarkom 24-igłowym. Quietwriter II (oznaczony też symbolem IBM 5202-001) drukuje z maksymalną szybkością 160 znaków/s w trybie przeglądowym (dratt), 100 zn/s w trybie NLQ i 80 zn/s z jakością korespondencyjną LQ (Letter Quality). Jakość druku w trybie przeglądowym niewiele różni się od druku korespondencyjnego innych drukarek - znak jest tworzony w matrycy 36x40 punktów! Możliwy jest też druk w trybie graficznym z rozdzielczością 240x360 punktów na cal.
Doskonałe parametry Quietwriter III zawdzięcza termicznej metodzie druku. Głowica drukarki zawiera 41 niezależnie pobudzanych termoelementów i drukuje jednokierunkowo przy użyciu specjalnej taśmy barwiącej, nasyconej barwnikiem termoczułym (na bazie wosku). Taśma wystarcza na wydrukowanie 385 tysięcy znaków przy druku normalnym lub 265 tysięcy znaków przy korespondencyjnym - odpowiada to objętościom 200 i 150 stron A4. Zestaw termoelementów trzeba wymieniać co ok. 3 miliony znaków, czyli ok. 1600 stron. W RFN taśma barwiąca kosztuje 42 DM, a zestaw termoelementów 93 DM - drukarka jest więc dość kosztowna w eksploatacji. Na szczęście wystarcza jej zwykły papier - ciągły z boczną perforacją lub kartki A4. Ze względu na metodę druku nie można, niestety, drukować kilku kopii jednocześnie.
Quietwriter III dysponuje czterema wbudowanymi krojami pisma: Courier 10, 12, 17,1 oraz Boldface PS. Dalsze 28 krojów można uzyskać stosując dodatkowe moduły generatorów znaków, z których każdy zawiera cztery rodzaje czcionek. W pamięci RAM drukarki znajduje się bufor o pojemności 3600 znaków, lecz nie można w niej definiować własnych czcionek (tzw. download fonts). Za dodatkową opłatą (ok. 400 DM) można natomiast kupić moduły pamięci RAM z podtrzymywaniem bateryjnym, które są umieszczane zamiast generatorów znaków i pozwalają dowolnie definiować jeden zestaw czcionek. Krój pisma zapisany w takim module jest utrzymywany przez 3 lata.
Cena drukarki Quietwriter III jest równie wysoka jak jej parametry - wynosi ok. 3500 DM, spore są również koszty eksploatacji. Nie jest to więc sprzęt dla amatorów - raczej dla dobrze sytuowanych zwolenników druku wysokiej jakości i ciszy w miejscu pracy. Computer Personalich **(JaW)**

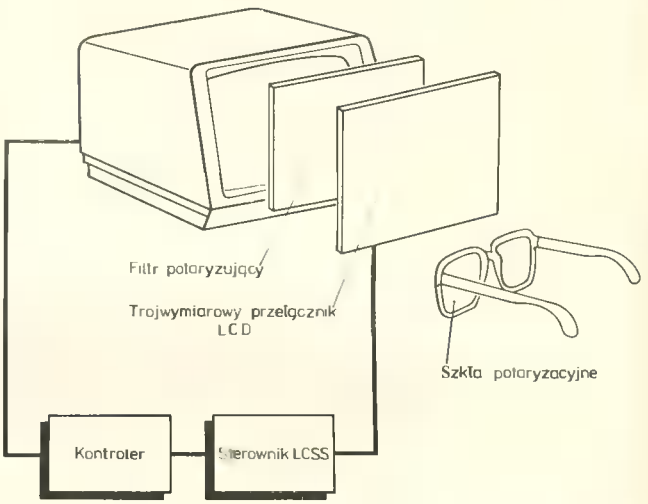
LAN i światłowody

Apple Talk, sieć lokalna Apple stworzona dla Macintosh'a, uzupełniona została o wersję wykorzystującą światłowody zamiast tradycyjnych połączeń kablowych. Światłowodowa sieć LAN może łączyć ponad 100 stacji, tzn. trzykrotnie więcej niż tradycyjna. Największa odległość między stacjami wydłużona została z ok. 300 m do ok. 1500 m. Poza tym sieć światłowodowa, niewrażliwa na pola elektromagnetyczne i wszelkie zakłócenia zewnętrzne, pozwala na szybszą i pozbawioną błędów transmisję danych. W niektórych zastosowaniach cenna jest też inna zaleta światłowodów - zabezpieczają one przed „włamaniem" się do systemu z zewnątrz - gdyż przesyłana informacja nie wywołuje pola elektromagnetycznego, które może być odbierane przy zastosowaniu odpowiednio czułych urządzeń
Producentem zarówno światłowodów, jak i całego sprzętu LAN jest Du Pont. Światłowodowa sieć ma modularną architekturę i składa się z zespołów przetwarzających sygnały elektryczne na optyczne (i odwrotnie) oraz wiązek światłowodowych. Budowa sieci polega na przyłączeniu wszystkich stacji bezpośrednio do konwerterów sygnałów i poprowadzeniu światłowodów. Sieć współpracuje z całym oprogramowaniem Macintosh'a bez konieczności wprowadzania zmian lub uzupełnień. Cały system jest łatwiejszy do zainstalowania niż tradycyjny dzięki małym wymiarom urządzeń i światłowodów, które obywają się bez ekranowania czy specjalnego zabezpieczania.
Dotychczas Du Pont - jeden z największych na świecie koncernów chemicznych - oferował dla przemysłu elektronicznego tworzywa i półprodukty. Wejście na rynek komputerowy z wyrobem finalnym, jakim jest opisana sieć LAN, jest wyrazem nowych tendencji marketingowych. Nawiasem mówiąc, Du Pont jest także jednym z największych na świecie użytkowników komputerów
DU PONT Information Service **(pc)**



Terminal 3D

Na targach w Hannoverze firma TEKTRONIX przedstawiła stereoskopowy, trójwymiarowy system projektowania CAD/CAM o symbolu 4126 z umieszczonym przed monitorem ciekłokrystalicznym stereoskopowym przełącznikiem - LCSS, mającym możliwość sekwencyjnego powtarzania obrazu. Jest on przełączany za pomocą sygnału synchronizującego tak, aby kolejne obrazy miały przeciwną polaryzację, a okulary do obserwowania obrazów stereoskopowych (z filtrami polaryzującymi) rozdzielają obrazy dla jednego i drugiego oka. Dzięki zastosowaniu LCSS okulary nie są połączone elektrycznie z systemem. Eliminuje to błędy widzenia oraz bóle głowy występujące po dłuższym czasie pracy. System 4126 ułatwia pracę przy projektowaniu skomplikowanych kompleksowych układów trójwymiarowych dla różnych dziedzin życia (mechanika, geofizyka, architektura, kartografia, ergonomia inżynierska). Wszystkie pakiety programowe techniki 3D dla modeli 4129 i 4128 mogą być użyte bez przeszkód dla 4126.
DESIGN & ELEKTRONIK **(rr)**



Zasada działania monitora trójwymiarowego

Nie ma jutra bez transputera

Sytuację technologii europejskiej najlepiej określa wypowiedź Iana Barron'a - twórcy transputera, jednego z najlepszych specjalistów w technologii półprzewodników: „być technologiem w Europie jest bardzo frustrujące - często poziom technologii jest znacznie wyższy niż w USA, ale nie ma pieniędzy na inwestycje".
Opinię tę potwierdza ośmioletnia historia wynalazku transputera i fabryki półprzewodników Inmos.
W 1978 r. brytyjski rząd Partii Pracy powierzył Barron'owi i dwóm inżynierom z USA uruchomienie w Inmos (Bristol) produkcji transputera. Chodziło jednocześnie o stworzenie szybko rozwijającej się nowoczesnej fabryki półprzewodników. 10 miesięcy później, zanim firma Inmos rozpoczęła produkcję, wybory w Wielkiej Brytanii wygrała Partia Konserwatywna z M. Thatcher na czele. „Żelazna Lady" zaczęła gwałtownie likwidować wszystkie inwestycje, które rozpoczął poprzedni rząd, dlatego postanowiła sprzedać również Inmos. Dopiero jednak w 1984 r. fabrykę kupiła firma Thorn EMI.
Europejscy nabywcy boją się rewolucyjnych rozwiązań technologicznych, takich jak transputer. Nikt w Europie nie ma zaufania do produkcji Inmos - firmy malutkiej w porównaniu z takimi gigantami, jak Motorola czy Intel. Mimo to Inmos od 1978 produkowała i sprzedawała transputery producentom komputerów na całym świecie ... z wyjątkiem brytyjskiego ICL.
Firma Thorn EMI jest jednak zbyt mała i nie ma dużego zaplecza finansowego do rozwoju nowych technologii, a w Europie nie może znaleźć inwestorów. Zainteresowane transputerami są natomiast Stany Zjednoczone i Japonia. IBM stosuje transputery w sterownikach monitorów kolorowych wysokiej rozdzielczości, japońska firma KDD będzie na bazie transputera produkować videotelefon: transputer stosowany jest w video kamerach - oczywiście przez producentów japońskich. Amerykanie i Japończycy chętnie zapłacą za technologię, ale nie interesuje ich rozwój fabryki półprzewodników w Bristolu. Dlatego ludzie tacy jak Iann Barron - jeśli chcą zobaczyć swoje genialne pomysły produkowane i wykorzystane - muszą, niestety, wyjechać z Europy. International Management **(rh)**

Młodszy brat SideKicka

Prezes Borlandu Philippe Kahn przedstawił w Londynie nowy produkt swojej firmy: SideKick Plus - ulepszoną i rozwiniętą wersję popularnego programu-pomocnika. Podobnie jak poprzednik, SideKick Plus rezyduje w pamięci komputera i pozwala korzystać z takich funkcji jak notatnik, kalendarz czy kalkulator. Jądro nowego programu zarządza różnymi przyłączanymi modułami. Wśród dostępnych obecnie modułów znajduje się 10 notatników, książka telefoniczna z funkcjami telekomunikacyjnymi, baza danych i kalkulator o bogatym zestawie funkcji.
Firma Borland oferuje ponadto pakiet rozwojowy, zachęcając innych producentów do tworzenia nowych modułów. Jeśli ten zamiar powiedzie się, Side Kick Plus może z „komputerowego pomocnika „przeobrazić" się w program rezydujący ponad warstwą systemu operacyjnego i konkurujący z takimi rozwiązaniami jak GEM czy Microsof Windows.
Your Computer **(JaW)**

BIOS jest oprogramowaniem systemowym na tyle zintegrowanym ze sprzętem, że z pozycji programisty może być traktowany jak element sprzętu. Omówienie bloków funkcjonalnych płyty głównej PC XT musi więc zostać uzupełnione opisem BIOS-a. Tradycyjnie publikowany schemat zastępujemy w tym numerze tablicą standardowych funkcji.

BIOS IBM PC

Zbigniew Guzik

Każdy system komputerowy powinien mieć chociaż minimalną pamięć stałą (ROM), zawierającą program ładujący z pamięci masowej system operacyjny. Pamięć ta, po załadowaniu systemu operacyjnego jest zwykle odłączana, a do dyspozycji użytkownika udostępniany jest cały obszar RAM. System operacyjny CP/M wymaga jedynie, aby program zawarty w pamięci stałej mógł odczytać pierwszy sektor dysku, w którym znajduje się program dalszej inicjalizacji komputera.

Koncepcja IBM PC i podstawowego systemu tego komputera - PC-DOS jest nieco inna niż w komputerach wcześniejszych generacji. Pamięć stała o pojemności 8 KB zawiera cztery części: inicjalizację systemu, program diagnostyczny, tablicę stałych systemowych i, co najważniejsze, zbiór podstawowych procedur wykorzystywanych do współpracy ze sprzętem.

Program zawarty w pamięci stałej IBM PC nosi nazwę BIOS (Basic Input/Output System) i choć jest zawarty w pamięci ROM ma cechy części Dyskowego Systemu Operacyjnego. Może on być wykorzystywany przez inny system operacyjny lub całkowicie pominięty. Na dyskietce systemowej znajduje się zbiór IBMBIO.COM, który jest rozszerzeniem procedur podstawowych i zawiera niezbędne dla danej konfiguracji komputera modyfikacje i dopełnienia. Zawarte w BIOS procedury obsługi są elementarne (np. odczyt sektora z dysku) i stanowią niejako warstwę izolacyjną pomiędzy wywołaniami funkcji wyższego poziomu a sprzętem. W ten sposób BIOS zapewnia niezależność sprzętową i umożliwia przenoszenie systemów operacyjnych i programów użytkowych między komputerami o różnej konfiguracji. Opracowana przez firmę Microsoft podstawowa wersja BIOS-a powstała w kwietniu 1981 r. - jest ona prawnie zastrzeżona.

BIOS w IBM PC dzieli całą dostępną przestrzeń adresową (1 MB) na trzy stałe obszary: pamięć operacyjną RAM (640 KB), pamięć monitorów ekranowych (128 KB) i pamięć stałą ROM (górne 256 KB). Najniższy obszar o pojemności około 1500 bajtów jest zarezerwowany przez BIOS oraz DOS i nie może być użyty do innych celów. Obszar ten składa się z tablic wektorów przerwań i zmiennych systemowych. Mapa przestrzeni adresowej IBM PC XT/AT została szczegółowo omówiona w **mikro**klanie 6/87.

Przerwania

Przerwania w systemach sterowanych procesorem 8086/88 mogą być zgłaszane wewnętrznie (w wyniku wykonania rozkazu INT lub w nieprawidłowych sytuacjach, np. dzielenie przez 0) i urządzenia takie, jak sterowniki czy interfejsy.

Każde z 256 możliwych przerwań identyfikowane jest przez numer, wykorzystywany jako indeks dla określenia miejsca w tablicy wektorów, zawierającego adres podprogramu obsługi przerwania. Po przyjęciu przerwania procesor zapisuje na stosie bieżącą zawartość rejestru CS, wskaźnika rozkazów IP i słowo stanu, umieszczając w rejestrach CS i IP dane z tablicy wektorów. Tablica wektorów przerwań, o rozmiarach 1 KB, umieszczona jest w najniższym obszarze przestrzeni adresowej. Każdy wektor składa się z dwubajtowego adresu segmentu kodu programu CS i dwubajtowego adresu przemieszczenia wewnątrz segmentu.

Pięć przerwań o najmniejszych numerach jest dedykowane sytuacjom specjalnym i sprzętowemu przerwaniu niemaskowalnemu (NMI). Przerwanie niemaskowalne jest generowane w IBM PC po wykryciu błędu parzystości zapisu w pamięci RAM i przez koprocesor arytmetyczny 8087. Następne 27 przerwań producent mikroprocesora zarezerwował na przyszłe zastosowania, ale projektanci IBM zlekceważyli to i resztę (27+224) adresów wykorzystują na przerwania sprzętowe i programowe. Na przerwania sprzętowe w IBM PC przeznaczone jest osiem numerów (8-FH), których kody dostarczane są przez sterownik 8259 w cyklu potwierdzania przyjęcia przerwania (INTA).

Tylko 16 numerów (10H...1FH), przeznaczonych jest na przerwania programowe, realizujące wywołania procedur podstawowej wersji BIOS-a. Ponieważ wektory przerwań programowych znajdują się w pamięci RAM, łatwo można zmienić wywoływaną funkcję przez zmianę adresu w tablicy wektorów. W ten sposób użytkownik może napisać swój program obsługi i umieścić go w dowolnym miejscu pamięci operacyjnej.

Dla realizacji wszystkich niezbędnych zadań BIOS-a potrzebna jest znacznie większa liczba odwołań. Przyjęto zasadę, że każde przerwanie przypisane jest jednemu konkretnemu typowi urządzenia i może wykonywać wiele niezależnych zadań, identyfikowanych przez kod, umieszczany w rejestrze AH. Wszystkie parametry, przekazywane między programem głównym a wywoływaną procedurą, przesyłane są za pośrednictwem rejestrów wewnętrznych procesora (8086/88 ma 14 rejestrów 16-bitowych). Przyjęto, że do przekazywania adresów odwołań do pamięci powinny być używane rejestry BX i ES. Ponadto wszystkie rejestry, z wyjątkiem AX, powinny zachować swoją pierwotną wartość. W tabeli zestawiono numery przerwań (w zapisie szesnastkowym) i odpowiadające im procedury obsługi.

Zerowanie systemu

Uruchamianie systemu (tzw. zimny start) inicjowane jest bądź przez załączenie zasilania, bądź też za pomocą przycisku RESET. Licznik rozkazów procesora zostaje wyzerowany, a rejestr segmentu kodu (CS) zapełniony jedynkami, wobec czego kod programu wprowadzany jest od adresu FFFF0H, znajdującego się w obszarze pamiąci stałej BIOS. Pod tym adresem znajduje się międzysegmentowy skok do części BIOS-a, gdzie wykonywany jest program diagnostyczny POST (ang, Power On Self Test). Składa się on z 14 testów systemu IBM PC: testu procesora 8086/88, sprawdzenia sumy kontrolnej pamięci ROM zawierającej BIOS, określenia konfiguracji przyłączonego sprzętu, testu pamięci RAM, sprawdzenia poprawności działania sterownika monitora ekranowego, klawiatury i dysków elastycznych oraz ich inicjalizacji, a także procedury rozszerzenia systemu. Sprawdzanie pamięci RAM odbywa się przez wykonanie pięciu różnych testów zapisu-odczytu, w których stosowane są różne kombinacje bitowe (00H, FFH, 55H, AAH i liniowa). Jest to bardzo czasochłonna operacja, która w zależności od rozmiaru zainstalowanej pamięci, może trwać od kilku sekund do kilku minut. Tzw. ciepły start omija te operacje, co znacznie skraca czas zerowania systemu.

Po prawidłowym wykonaniu wymienionych programów, wywoływane jest przerwanie numer 19H (bootstrap), którego procedura obsługi wczytuje do pamięci pierwszy sektor z zerowej ścieżki na dysku. Sterowanie przekazane jest do wczytanego programu, który ładuje system operacyjny. Gdy w urządzeniu nie ma dyskietki lub jej zawartość nie może być odczytana, wywołany jest BASIC,

PRZERWANIA SYSTEMU [illegible]

parametry wejściowe	parametry wyjściowe

INT 0H	Błąd dzielenia przez zero	z procesora
INT 1H	Praca krokowa	z procesora
INT 2H	Przerwanie niemaskowalne	
INT 3H	Punkt wstrzymania	z procesora
INT 4H	Nadmiar	z procesora
INT 5H	Wydruk zawartości ekranu monitora	
INT 5H	Zarezerwowane	
INT 7H	Zarezerwowane	
INT 8H	Zegar czasu rzeczywistego	z 8259A
INT 9H	Klawiatura	z 8259A
INT AH	Zarezerwowane	z 8259A
INT BH	Sterownik transmisji szeregowej COM 2	z 8259A
INT CH	Sterownik transmisji szeregowej COM 1	z 8259A
INT DH	Sterownik sztywnego dysku	z 8259A
INT EH	Sterownik dysków elastycznych	z 8259A
INT FH	Interfejs Centronics (drukarka)	z 8259A

INT 10H Obsługa monitorów ekranowych

parametry wejściowe	parametry wyjściowe
wybór trybu pracy	(1)
AH = 00H	
AL=tryb pracy	
ustawienie rozmiaru kursora	(2)
AH=01H	
CH=górna linia	
CL=dolna linia	
ustawienie pozycji kursora na ekranie	
AH=02H	
BH=numer strony	
DH=rząd	
DL=kolumna	
odczytanie pozycji kursora	
AH=03H	CH=górna linia
BH=numer strony	CL=dolna linia
	DH=rząd
	DL=kolumna
odczytanie pozycji pióra świetlnego	
AH=04H	AH=sygnał wyzwalania
	BX=kolumna punktów
	CH=wiersz punktów
	DH=rząd znakow
	DL=kolumna znaków
ustawienie aktywnej strony	
AH=05H	
AL=numer strony	
przewijanie okna w górę	
AH=05H	
AL=liczba dod. wierszy	
BH=atrybut dod. wierszy	
CH=górny rząd	
CL=lewa kolumna	
DH=dolny rząd	
DL=prawa kolumna	
przewijanie okna w dół	
AH=07H	
AL=liczba dod. wierszy	
BH=atrybut dod. wierszy	
CH=górny rząd	
CL=lewa kolumna	
DH=dolny rząd	
DL=prawa kolumna	
odczytanie [illegible] i atrybutu	
AH=08H	AH=znak
BH=numer strony	AL=atrybut
zapis znaku i atrybutu	
AH=09H	
AL=znak	
BH=numer strony	
BL=atrybut	
CX=liczba znak. do zapis.	
zapis znaku	
AH=0AH	
AL=znak	
BH=numer strony	
BL=kolor (jeżeli tryb graf.)	
CX=liczba znak. do zapis.	
ustawienie palety kolorów	
AH=0BH	
BH=identyfikator palety	
BL=kolor z wybran. palety	
zapis punktu	
AH=0CH	
AL=kolor	
CX=kolumna	
DL=wiersz	
odczyt punktu	
AH=0DH	AL=odczytany kolor
CX=kolumna	
DL=wiersz	
symulacja pracy terminala	
AH=0EH	
AL=znak	
BL=kolor (jeżeli tryb graf.)	
odczyt aktualnego trybu pracy	
AH=0FH	AH=szerokość (w znak.)
AL=tryb pracy	
BH=numer strony	
zapis łańcucha znaków bez zmiany pozycji kursora AT	
AH= 13H	
AL= 00H	
BL= atrybut	
BH= numer strony	
DX= pozycja kursora	
CX= długość łancucha	
ES:BP= wskaźnik	
początku łańcucha	
zapis łańcucha znaków z przesunięciem kursora AT	
AH=13H	
AL= 01H	
BL= atrybut	
BH = pozycja kursora	
CX = długość łańcucha	
ES:BP = wskaźnik	
początku łańcucha	
zapis łańcucha znaków bez zmiany pozycji kursora AT [illegible]	
AH = 13H	
AL =02H	
BH =numer strony	
DX =pozycja kursora	
CX =długość łancucha	
ES:BP =wskaźnik	
początku łańcucha	
zapis łańcucha znaków z przesunięciem kursora AT	
AH =13H	
AL =03H	
BH =numer strony	
DX =pozycja kursora	
CX =długość łancucha	
ES:BP =wskaźnik	
początku łańcucha	

INT 11H odczyt konfiguracji dołączonego wyposażenia (4)

AX =lista	

INT 12H odczyt rozmiarów zainstalowanej pamięci

AX =rozmiary pamięci	

INT 13H obsługa dysków elastycznych (5)

parametry wejściowe	parametry wyjściowe
inicjalizacja systemu dyskowego	
AH =00H	
odczyt statusu ostatnio wykonywanej operacji dyskowej	
AH =01H	AL =kod statusu
odczyt sektorów	
AH =02H	CF =znacznik prawidłowej oper
AL =liczba sektorów	AH =kod [illegible]
CH =numer ścieżki	AL =liczba przeczyt. sektorów
CL =numer sektora	
DH =numer głowicy	
DL =numer stacji	
ES:BX =wskaźnik bufora	
zapis sektorów	
AH =03H	CF =znacznik prawidł. operacji
AL =liczba sektorów	AH =kod statusu
CH =numer ścieżki	AL =liczba zapisanych sektorów
CL =numer sektora	
DH =numer głowicy	
DL =numer stacji	
ES:BX =wskaźnik bufora	
weryfikacja sektorów	
AH =04H	CF =znacznik prawidł. operacji
AL =liczba sektorów	AH =kod statusu
CH =numer ścieżki	AL =liczba zweryfik. sektorów
CL =numer sektora	
DH =numer głowicy	
DL =numer stacji	
formatowanie sektorów	
AH =05H	CF =znacznik prawidł. operacj
AL =liczba sektorów	AH =kod statusu
CH =numer ścieżki	AL =liczba przeczyt. sektorów
CL =numer sektora	
DH =numer głowicy	
DL =numer stacji	
ES:BX =wskaźnik listy	
4 bitowych pol adresowych	
odczyt parametrów aktualnej stacji dyskowej	
AH =08	DL =liczba stacji dyskowych
	DH =liczba stron
	CL =liczba sektorów
	CH =liczba scieżek
	CF =znacznik powodz. operacji
	AH =kod statusu
inicjalizacja tablic bazowych obu sztywnych dysków	
AH =09	CF =znacznik prawidł. operacj
	AH =kod statusu
odczyt d[illegible] liczby [illegible]	
AH =0A	CF =znacznik prawidł. operacj
DL =identyfikator dysku	AH =kod statusu
DH =numer głowicy	
CH =numer cylindra	
CL =numer sektora	
ES:BX =wskaźnik bufora	
zapis dużej liczby sektorów	
AH =0B	CF =znacznik prawidł. operac
DL =identyfikator dysku	AH =kod [illegible]
DH =numer głowicy	
CH =numer cylindra	
CL =numer sektora	
ES:BX =wskaźnik bufora	
przesunięcie głowicy	
AH =0C	CF =znacznik prawidł. operac
DL =identyfikator dysku	AH =kod statusu
DH =numer głowicy	
CH =numer cylindra	
alternatywna inicjalizacja systemu dysków	
AH =0D	CF =znacznik prawidł. operac
DL =identyfikator dysku	AH =kod statusu
test czy dysk jest gotowy	
AH =10	CF =znacznik prawidł. operacj
DL =identyfikator dysku	[illegible] =kod statusu
rekalibracja dysku	
AH =11	CF =znacznik prawidł. operac
DL =identyfikator dysku	AH =kod [illegible]
diagnostyka sterownika	
AH = [illegible]	CF =znacznik prawidł. operac
DL =identyfikator dysku	AH =kod statusu
odczyt typu dysku AT	(6)
AH =15	AH =typ dysku
DL =identyfikator dysku	CX,DX =liczba [illegible] bitowych sekt. gdy AH = 3
odczyt statusu dysku AT	
AH =16	DL =numer zamienion. dysku
	AH =status zamienion. dysku
ustawienie statusu stacji dysków	[illegible]
AH =17	
AL =typ statusu	

INT 14H Obsługa intrfejsu szeregowego (8)

parametry wejściowe	parametry wyjściowe
inicjalizacja interfejsu	
AH =00	AX =status portu
AL =parametry	
wysłanie jednego znaku przez interfejs	
AH =01	[illegible] =status portu
AL =znak	
DX =numer portu	
odbiór jednego znaku	
AH =02	AH =status operacji
DX = numer portu	AL= znak
odczyt statusu portu	
AH=03	AX=status portu

parametry wejściowe	parametry wyjściowe

INT 15H Obsługa magnetofonu kasetowego

włączenie silnika magnetofonu
AH = 00
wyłączenie silnika magnetofonu
AH = 01
czytanie bloku danych
AH = 02 — CF = sygnał błędu
CX = liczba bitów — DX = liczba przeczyt. bitów
ES:BX = wskaźnik obszaru danych — ES:BX = wskaz. ostatnio przecz bitu
zapis bloku danych
AH = 03 — ES:BX = wskaź. ostatnio przecz bitu
CX = liczba bitów
ES:BX = wskaźnik obszaru danych

INT 15H Dodatkowe funkcje dla AT

otwarcie kanału do urządzenia AT
AH = 80
BX = identyfikator urządzenia
CX = identyfikator procesu
zamknięcie kanału do urządzenia AT
AH = 81
BX = identyfikator urządzenia
CX = identyfikator procesu
zakończenie obsługi urządzenia AT
AH = 82
BX = identyfikator urządzenia
oczekiwanie na zdarzenie AT
AH = 83
AL = 0 - ustawienie okresu
1 - rezygnacja z oczekiwania
ES:BX = wskaź. obszaru pamięc
CX,DX = okr. czasu do oczekiw.
obsługa manipulatora AT
AH = 84 — AL = ustawienie przel.
DX = 00 - odczyt. ustaw. przel. — AX = wartość A(x)
DX = 01 - odczyt. sygnałow — BX = wartość A(y)
CX = wartość B(x)
DX = wartość B(y)
obsługa klawisza System Request AT
AH = 86 — AL = 0 wcisnięto klawisz
AL = 1 puszczono klawisz

parametry wejściowe	parametry wyjściowe

opóźnienie AT
AH = 86
CX,DX - czas opóźnienia
przesłanie bloku danych między obszarami pamięci AT
AH = 87
CX = liczba słów do przesłan.
ES:SI = wskaźnik tablicy
odczytanie wielkości rozszerzonej pamięci AT
AH = 86 — AX = liczba ciągłych 1K bloków pamięci rozpocz. się powyżej 1024KB
przełączenie procesora w tryb wirtualny AT
AH = 89 — AH = 0 jeżeli oper. się powiodła
pętla oczekiwania na urządzenie
AH = 90
AL = kod typu
ustawienie znacznika i dokończenie obsługi przerwania
AH = 91
AL = kod typu

INT 16H obsługa klawiatury

odczyt znaku
AH = 00 — AH = numer klawisza
AL = kod znaku
odczyt stanu bufora klawiatury
AH = 01 — ZF = brak znaku
AH = numer klawisza
AL = kod znaku
odczyt klawiszy sterujących (9)
AH = 02 — AL = bity klawiszy

INT 17H obsługa drukarki

wysłanie jednego bajtu do drukarki (10)
AH = 00 — AH = kod statusu
AL = znak
inicjalizacja drukarki
AH = 01 — AH = kod statusu
odczyt statusu drukarki
AH = 02 — AH = kod statusu

INT 18H wywołanie BASICa z pamięci stałej
INT 19H gorący restart

parametry wejściowe	parametry wyjściowe

INT 1AH obsługa zegara

odczytanie stanu zegara
AH = 00 — AL = przepełnienie dobowe
CX = bardziej znaczący bajt liczn.
DX = mniej znaczący bajt licznika
ustawienie aktualnego czasu
AH = 01
CX = bardziej znaczący bajt liczn.
DX = mniej znaczący bajt licznika
odczytanie zegara czasu rzeczywistego AT
AH = 02 — CH = godziny
CL = minuty
DH = sekundy
ustawienie zegara czasu rzeczywistego AT
AH = 03
CH = godziny
CL = minuty
DH = sekundy
DL = 1 jeżeli czas letni
odczyt daty z zegara czasu rzeczywistego AT
AH = 04 — DL = dzień
DH = miesiąc
CL = rok
CH = wiek (19 lub 20)
ustawienie daty w zegarze czasu rzeczywistego AT
AH = 05
DL = dzień
DH = miesiąc
CL = rok
CH = wiek (19 lub 20)
ustawienie budzika
AH = 06
CH = godziny
CL = minuty
DH = sekundy
wyłączenie budzika
AH = 07

INT 1BH obsługa procedury wyw. naciśnięciem Ctrl i Break
INT 1CH obsługa procedury wywoływanej [illegible] przerwanie zegarowe INTR 08
INT 1DH wektor tablicy inicjalizacyjnej monitorów ekran.
INT 1EH wektor bloku parametrów inicjalizacji sterownika dysków
INT 1FH wektor tablicy wzorów graficznych znaków ASCII

Uwagi
(1) tryb pracy monitora
0A 640x200 grafika [illegible] kolory
09 320x200 grafika 16 kolorów
08 160x200 grafika 16 kolorów
07 80 x 25 text czerń i biel
06 640x200 grafika czerń i biel
05 320x200 grafika 4 stopnie jasności
04 320x200 grafika 4 kolory
03 80 x 25 text 16 stopni jasności
02 40 x 25 text 16/8 kolorów
01 40 x 25 text 16 stopni jasnosci

(2) CGA–linie 0-7
[illegible]

(3) Każdy znak może mieć inny atrybut
(4)znaczenie poszczególnych bitów w liście
14,15 liczba dołączonych drukarek
13 = obecnośc drukarki szeregowej
[illegible] = obecność portu manipulatora
09,10,11 = liczba interfejsow szeregowych
[illegible] = obecność kanału DMA
[illegible] = jest, [illegible] = brak
06,07 = liczba stacji dysków elastycznych
04,05 = początkowy tryb wideo
[illegible] = nie uzywany
01 = 40 x 25 kolorowy
10 = 80 x 25 kolorowy
11 = 80 x 25 czarno-biały
02,03 = pamięc RAM na płycie głównej [illegible] blokach po [illegible]
01 = koprocesor arytmetyczny
00 = stacja dyskow elastycznych

(5)kody statusu
00 = operacja wykonana prawidłowo
01 = błędny rozkaz
02 = brak znaku adresu
[illegible] = próba zapisu na zabezpieczoną dyskietkę
04 = nie znaleziono sektora
06 = dyskietka została wyjęta
06 = błąd sprzętowy kanału DMA
09 = przekroczenie granicy adresowej obszaru DMA
[illegible] = błąd sumy kontrolnej
20 = błąd sprzętowy sterownika dysku NEC 765
40 = błąd pozycjonowania głowicy
80 = przekroczony czas operacji

(6) typy dysków
AH
0 = nie zainstalowana stacja dysków
1 = stacja dyskow elastycznych, brak sygnału zmiany
2 = stacja dysków elastycznych, obecnosć sygnału zmiany
3 = dysk stały

(7) typy statusu
0 = nie używany
[illegible] = dysk 360K w stacji 360K
2 = dysk 360K w stacji 1,2M
3 = dysk 1,2M w stacji 1,2M

(8) parametry inicjalizacji interfejsu szeregowego
00,01 = długość słowa: 10 = 7 bitów, 11 = 8 bitów,
02 = bity stopu: 0 = 1, 1 = 2
03,04 = rodzaj kontroli parzystości:
00,01 = brak, 11 = parzystość, 01 = nieparzystość,
05,06,07 szybkość transmisji:
[illegible] = 110;
[illegible] = 150;
010 = 300;
011 = 600;
100 = 1,200;
101 = 2,400;
110 = 4,800;
111 = 9,600;

status portu
bity [illegible] AH - status operacji
[illegible] = przekroczenie czasu
[illegible] = pusty rejestr przesuwny, nadawczy
[illegible] = pusty rejestr buforujący, nadawczy
[illegible] = stwierdzono przerwę w transmisji
03 = błąd ramki
02 = błąd parzystości
01 = błąd zgubienia znaku
00 = układ gotowy
bity w AL - status linii
07 = obecnośc nosnej
06 = żądanie odbioru z linii telefonicznej
06 = gotowość modemu do nadawania
04 = gotowość modemu do odbierania
03 = zmiana stanu wskaźnika obecności nośnej
02 = zmiana stanu wskaźnika żądania odbioru z linii telefonicznej
[illegible] = zmiana stanu wskaźnika gotowosci modemu do nadawania
00 = zminay stanu wskaźnika gotowości modemu do odbioru

(9) bity klawiszy sterujących
7 - aktywny INSERT
5 - aktywny CAPS LOCK
5 - aktywny NUM LOCK
4 - aktywny SCROLL LOCK
3 - wciśnięty ALT
2 - wciśnięty CTRL
1 - wciśnięty lewy SHIFT
0 - wciśnięty prawy SHIFT

(10) kody statusu dla obsługi drukarki
7 = 1 - nie zajęta
6 = 1 - potwierdzenie
5 = 1 - brak papieru
[illegible] = 1 - wybranie drukarki
3 = 1 - błąd wejścia-wyjścia
1,2 - nie używane
0 - przekroczenie limitu czasu obsługi

znajdujący się również w pamięci stałej. W wielu konfiguracjach IBM PC, a szczególnie w wersjach dalekowschodnich, pamięć stała nie zawiera języka BASIC i do uruchomienia systemu potrzebna jest dyskietka lub dysk sztywny zawierające system operacyjny. W oryginalnych komputerach od adresu fizycznego FFFF5H w pamięci ROM przechowywane są dane o miesiącu, dniu i roku udostępnienia danej wersji BIOS-a.

Procedury obsługi urządzeń zewnętrznych

BIOS zawiera podstawowe procedury obsługi pięciu urządzeń peryferyjnych: monitora ekranowego, dysku elastycznego, klawiatury, interfejsu transmisji szeregowej i drukarki. Mankamentem oprogramowania zawartego w BIOS-ie jest brak standardowych procedur generacji sygnału akustycznego. Programy takie użytkownik musi wprowadzić sam.
Klawiatura jest chyba jedną z najlepiej zaprojektowanych części komputera IBM PC. Zawiera komputer jednoukładowy (8048), którego zadaniem jest śledzenie wszystkiego, co dzieje się z klawiaturą i przekazywanie tego przez synchroniczne łącze szeregowe za pośrednictwem przerwania sprzętowego o numerze 9H. Sygnalizowane jest każde naciśnięcie i każde zwolnienie klawisza, niezależnie od tego, czy jest to klawisz sterujący czy zwykły. Standardowa klawiatura zawiera 83 klawisze, których numery (a nie kody ASCII) przekazywane są do BIOS-a. W przypadku naciśnięcia klawisza najbardziej znaczący bit przekazywanego bajtu zostaje wyzerowany, natomiast w momencie zwolnienia - bit ten jest ustawiany w stan 1. Przekazywane z klawiatury dane są umieszczane w specjalnym buforze, co pozwala na wprowadzenie ich z wyprzedzeniem. Klawiatura ma siedem klawiszy sterujących, trzy o charakterze monostabilnym (Shift, Ctrl, Alt), a pozostałe cztery (CapsLock, NumLock, ScrollLock i Ins) o charakterze bistabilnym. Od nich zależy interpretacja przesyłanych danych przez program obsługujący przerwanie sprzętowe. Klawiatura sygnalizuje również pewne sytuacje specjalne. Jednoczesne naciśnięcie klawiszy Ctrl-Alt-Del powoduje wyzerowanie systemu. Użycie klawisza PrtSc powoduje wydrukowanie zawartości ekranu. Kombinacja Ctrl-NumLock zawiesi wykonywanie bieżącego programu, aż do momentu naciśnięcia klawisza innego niż NumLock. Kombinacja Ctrl-Break inicjuje puste przerwanie programowe 1BH, które użytkownik może wykorzystać na wprowadzenie do systemu dowolnej nowej funkcji.
IBM PC może wykorzystywać dyski elastyczne i sztywne, aczkolwiek w podstawowych procedurach BIOS-a przewidziana została tylko obsługa dysku elastycznego. Po zainicjowaniu operacji dyskowej, układ sterownika dysku (NEC 765) przejmuje funkcje sterowania i zaczyna działać automatycznie. Zakończenie operacji sygnalizowane jest przez przerwanie sprzętowe (EH). Ze względu na zawodność takiego nośnika informacji, jakim jest dyskietka, w razie błędu zapisu lub odczytu procedury dyskowe są powtarzane. Maksymalna liczba powtórzeń ustalana jest w czasie inicjalizacji systemu. Jeżeli w systemie zainstalowany jest dysk sztywny, wówczas - po inicjalizacji - procedury BIOS-a używają przerwania o numerze 40H, a numer 41H przeznaczony jest na wskaźnik bloku parametrów dysku sztywnego.
W BIOS-ie przewidziana jest obsługa dwóch typów sterowników video: kart CGA i MGA. Zdobywająca popularność karta EGA, po autoinicjalizacji, wykorzystuje przerwanie o numerze 42H, a numer 43H stosowany jest jako wskaźnik parametrów inicjalizacyjnych.

Możliwości rozbudowy

BIOS może być rozszerzany w sposób automatyczny. Program diagnostyczny POST analizuje zawartość obszaru adresowego pamięci stałej ROM i poszukuje programów, które następnie instalują się same. W ten sposób możliwe jest proste przyłączanie dodatkowych sterowników urządzeń peryferyjnych (np. sterownika dysków sztywnych lub monitora EGA).
Przeszukiwanie pamięci ROM odbywa się w 2-kilobajtowych blokach. Nowy moduł ROM uważany jest za poprawny, jeżeli pierwszy bajt w bloku (o numerze 0) zawiera wartość 55H, a bajt 1 - wartość AAH. Kolejny bajt bloku zawiera informację o rozmiarach przyłączonego modułu ROM, wyrażoną w liczbie jednostek 512-bajtowych. Po tych trzech bajtach znajduje się kod programu, do którego, za pomocą wywołania międzysegmentowego (far call), przekazywane jest sterowanie. Wywołany program inicjalizuje dany sterownik, integruje go z całością systemu i ustawia odpowiednie wektory przerwań, po czym, za pomocą międzysegmentowego powrotu, przekazuje sterowanie ponownie programowi BIOS. Na rozszerzenia systemowe przewidziane są w tablicy wektorów indeksy o numerach powyżej 40H.

Wykorzystanie BIOS-a

Za system zgodny z IBM PC przyjęto uważać taki, który realizuje programowe wywołania BIOS-a zgodnie ze specyfikacją (tab.). Jednak mimo spełnienia tego warunku zdarza się, że nie wszystkie programy działają w takim systemie. Wynika to z faktu, że nie zawsze do realizacji żądanych funkcji wykorzystywane jest pośrednictwo BIOS-a (dla uzyskania większej szybkości).
System kompatybilny z IBM PC to taki, który, oprócz spełnienia warunku właściwej realizacji wywołań INT, zapewnia identyczne adresy portów we-wy i identyczne funkcje dla ich poszczególnych bitów. Ponadto muszą zostać zapewnione takie same adresy i interpretacja zmiennych systemowych. Zachowane muszą też być formaty danych na zewnętrznych nośnikach informacji oraz sposób przesyłania i upakowania informacji. Dla realizacji programów szczególnie wrażliwych na niekompatybilność komputera konieczne jest zachowanie identycznych adresów dla poszczególnych funkcji BIOS-a oraz zapewnienie takich samych parametrów wejściowych i wyjściowych.
Dla uniknięcia kłopotów z przenoszeniem programów, procedury BIOS-a należy wywołać przez przerwania systemowe i unikać używania adresów bezwzględnych programów obsługi (adresy te mogą różnić się w kolejnych modelach komputera). Drugą zasadą jest możliwe minimalne modyfikowanie środowiska systemowego. Zmiany stanów dla bitów portów we-wy powinny dotyczyć jedynie tych, które są istotne dla działania danego programu, a po zakończeniu realizacji procedury ich stan powinien być odtwarzany.
Ze szczególną ostrożnością należy modyfikować zawartość tablicy wektorów i zmiennych systemowych, pamiętając, że działamy w środowisku sterowanym przez przerwania. Z tego względu, gdy modyfikacje nie są już potrzebne należy przywrócić pierwotny stan tablic. Wyjątkowo uważnie należy traktować również obsługę przerwań sprzętowych. Nierozważny krok może spowodować utratę wartościowych zbiorów lub, co gorsza, uszkodzenie drogiego urządzenia.

Zbigniew Guzik

MikroLaur

Koło Użytkowników Mikrokomputerów Profesjonalnych Przy Radzie Stołecznej NOT Przygotowując konferencje PC-Standard'87 i Komputer'88 organizuje Konkurs MIKRO-LAUR'87 w zakresie nowoczesnych rozwiązań sprzętu mikrokomputerowego oraz oprogramowania użytkowego. W konkursie mogą wziąć udział osoby prawne i twórcy indywidualni, zgłaszając swe prace do 31 grudnia 1987r. Powinny one odznaczać się szczególnymi walorami w zakresie: funkcjonalności i wartości użytkowej, nowoczesności i innowacyjności, jakości i niezawodności działania. Prace wraz ze zgłoszeniem zawierającym komplet wymaganych danych należy kierować pod adresem:
Ośrodek Doskonalenia Kadr Technicznych RS NOT, 02-361 Warszawa, ul. Przemyska 11a.
Wszelkich informacji udziela sekretarz organizacyjny konferencji i konkursu p. Dariusz Kołakowski, tel. W-wa 17-63-53.

Zabezpieczanie programów

Obowiązujący w Polsce stan prawny - nie zmieniany od wielu lat - zmniejszył koszty rozwoju techniki mikrokomputerowej (nawet państwowe instytucje posługują się skopiowanymi programami - kupowany jest głównie sprzęt), ale też niemal całkowicie sparaliżował rozwój produkcji oprogramowania krajowego. Zanim wprowadzone zostaną zmiany w przepisach, pozostaje stosowanie zabezpieczeń. Być może opisane techniki zainspirują polskich programistów i umożliwią ochronę dorobku. Choć tekst ten może ułatwić łamanie zabezpieczeń, uważamy jednak, iż najważniejsza jest obrona przed masowym kopiowaniem; na uzdolnionych „hackers'ów" jak dotąd nie wymyślono jeszcze sposobu.

Metoda błędnych sektorów

Najprostszą metodą zabezpieczania dyskietki jest utworzenie tzw. błędnych sektorów, w których suma kontrolna nie odpowiada zapisanym danym. Typowy program kopiujący po napotkaniu takiego sektora zatrzyma się, wyświetlając komunikat o błędzie, albo też (zakładając, że i błędne dane mogą być potrzebne) wprowadzi prawidłową sumę kontrolną. Stwierdzenie braku błędu odczytu dla określonego sektora spowoduje wstrzymywanie programu przy próbie jego realizacji.

Metoda ta jest coraz rzadziej stosowana, bowiem zabezpieczenie można łatwo ominąć dzięki programom kopiującym bit po bicie. W jej udoskonalonej wersji używa się specjalnych urządzeń usuwających fragment nośnika magnetycznego. Powoduje to powstanie błędnego sektora, którego nie można skopiować za pomocą konwencjonalnego sprzętu. W zabezpieczonym programie realizowany jest zapis danych w uszkodzonym sektorze, a następnie próba ich odczytu. W trakcie odczytu fizycznie uszkodzonego obszaru utraci się nie tylko bity danych, ale też sygnał synchronizacji dane odczytane w pozostałej części sektora będą przypadkowe. Jeśli w sektorze nie występuje fizyczne uszkodzenie nośnika, to zapisana informacja zostanie odczytana prawidłowo, a to pozwala stwierdzić skopiowanie dyskietki.

Metoda dodatkowych ścieżek

Zmiana formatu dyskietki w taki sposób, że ma ona więcej ścieżek, niż przyjęto w standardzie (rys. 2), uniemożliwia kopiowanie konwencjonalnymi metodami. Sterownik dysków może ustawiać głowicę na ścieżce 40, 41, a nawet 42 i następnych, a także zapisywać i odczytywać dane z tych ścieżek, natomiast typowy program kopiujący czyta i zapisuje (w przypadku IBM PC) tylko 40 ścieżek. Niestety, opisana metoda nie może być stosowana we wszystkich wypadkach, gdyż wiele napędów dysków jest tak skonstruowanych, że nie można w nich ustawić głowicy za 39 ścieżką, a niektóre mogą ulec nawet uszkodzeniu.

Metoda dodatkowych sektorów

Na standardowo sformatowanej dyskietce IBM PC ścieżka ma 8 lub 9 sektorów i tyle czytają typowe programy kopiujące. Producenci oprogramowania umieszczają ich 10 lub nawet 11, skracając przerwy między sektorami (rys. 3a). Zabezpieczany program stwierdza oryginalność dyskietki sprawdzając obecność dodatkowego sektora (może być w nim informacja niezbędna do realizacji programu). Dodatkowe sektory nie różnią się od pozostałych - mogą być kopiowane przez program, w którym przewidziano tego typu zabezpieczenia.

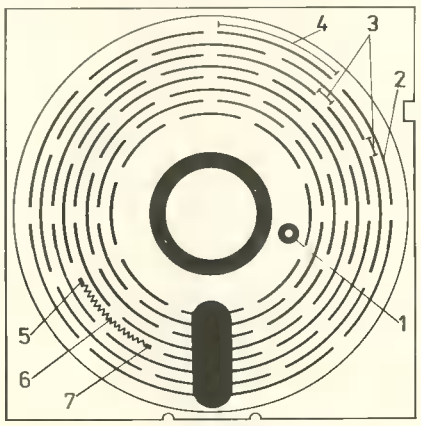
Sposobem oszukiwania takich programów jest technika nadmiarowych sektorów (rys. 3b). Polega ona na umieszczeniu na ścieżce samych nagłówków sektorów - bez danych. Ponieważ programy kopiujące działają na zasadzie rezerwowania dla poszczególnych sektorów niezbędnego obszaru pamięci w specjalnym buforze, więc zostanie on szybko zapełniony i ścieżka nie zostanie przekopiowana w całości. W programach kopiujących wprowadzane są jednak udoskonalenia, pozwalające na wykrycie i zignorowanie fałszywych sektorów.

Metoda słabych bitów

Technika słabych bitów polega na zapisie bitów reprezentujących stan pośredni między zerem a jedynką. Za każdym razem, gdy napęd dysków odczyta sektor zawierający słabe bity, otrzymywane są nieco inne dane (rys. 4). Skopiowanie takiego sektora w standardowym napędzie spowoduje zapis „normalnych" bitów. Odczytując ten sam sektor kilkakrotnie i porównując go, procedury sprawdzające mogą stwierdzić, czy dyskietka zawiera wersję oryginalną, czy też kopię. Jednak i standardowy napęd dysków można zmusić do symulacji słabych bitów. W tym celu należy odpowiednio szybko załączać i wyłączać układ zapisu w trakcie rejestracji danych.

Metoda wyrównywania sektorów

W przypadku normalnie zapisanej dyskietki do odczytania ścieżki potrzebna jest informacja o jej początku. Programy kopiujące zapisują ścieżkę na dyskietce tak, jak ją odczytały, nie dbając o to, aby wszystkie sektory zaczynały się w tym samym miejscu. Sprzęt służący do produkcji duplikatów może jednak zapisać dyskietkę w tak by wszystkie odpowiadające sobie numerami sektory na kolejnych ścieżkach zaczynały się w tym samym miejscu dyskietki (rys. 5). Program zabezpieczający może sprawdzić takie „wyrównanie" przez odczytanie zadanego sektora, przesunięcie głowicy na następną ścieżkę, odczekanie ściśle określonego czasu i odczytanie numeru następnego sektora, jaki napotka. Znając szybkość obrotową napędu dysków, można dokładnie przewidzieć, jaki numer powinien mieć napotkany sektor.



1. Standardowa dyskietka bez zabezpieczeń
1. otwór indeksowy (znacznik, względem którego jest ustalany początek pierwszego sektora na ścieżce); 2- ścieżka; 3- sektor; 4- przerwy między sektorami; 5- suma kontrolna kończąca sektor; 6-dane; 7- nagłówek zawierający numer sektora

Niekopiowalne dyskietki

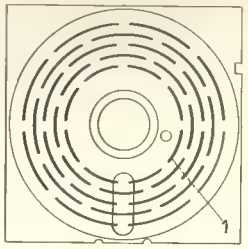
Zapisane w typowy sposób dyskietki mają ścieżki o kształcie koncentrycznym. Jeżeli jednak podczas procesu zapisywania danych głowica będzie przesuwana, powstanie tzw. zapis spiralny (rys. 6), który może być odczytywany przez standardowy napęd dysków, nie może być natomiast przez niego skopiowany.

Jedną z najczęściej stosowanych technik jest metoda supersektorów (rys. 7), wymagająca profesjonalnych urządzeń zapisujących, stosowanych przez producentów oprogramowania. Metoda polega na tworzeniu bardzo dużych sektorów (np. jeden sektor na ścieżce). Standardowy sterownik napędu dyskow może odczytać taki sektor, ale nie jest w stanie go zapisać. Dyskietkę oryginalną od kopii można odróżnić dzięki umieszczeniu specjalnych danych w przerwach między sektorami i odczytywanie całości jako jednego supersektora. Ponieważ standardowy układ sterowania nie może umieszczać danych w przerwach między sektorami, więc dodatkowa informacja znajdująca się w tym miejscu będzie przy kopiowaniu stracona. Aby uporać się z tym zabezpieczeniem firma Central Point Software Inc. (producent Copy II PC) opraco-

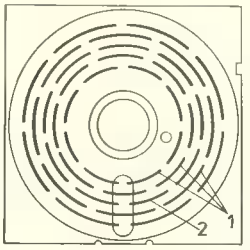
wała specjalną płytę interfejsu (przyłączaną między napędy dysków ▪ sterownik), która pozwala kopiować z dysku wszystko, co może być uznane za informację.
W standardowych napędach dysków wykorzystuje się stosunkowo wąskie szczeliny w głowicach, aby zminimalizować interferencje między danymi zapisywanymi na sąsiednich ścieżkach, natomiast urządzenia stosowane przez producentów mają szczeliny w głowicach szerokie, zdolne do równoległego umieszczenia danych na dwu lub trzech ścieżkach równocześnie. Na przykład mogą one zapisać dwie sąsiednie ścieżki i przestrzeń między nimi, tworząc bardzo szeroką ścieżkę (rys. 8). Przestrzeń między ścieżkami nie będzie skopiowana przez standardowy napęd. Przy przesuwaniu głowicy z jednej ścieżki na drugą nad wolną przestrzenią - nastąpi błąd odczytu.
Opisane metody eliminują kopiowanie zabezpieczonych dyskietek. Nie oznacza to jednak, że nie można złamać takich zabezpieczeń - wystarczy ominąć procedury sprawdzające.

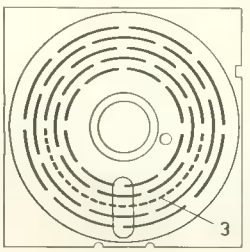
Procedury sprawdzające

Sposób omijania zabezpieczeń jest na pozór prosty. Jeśli system ochrony działa na zasadzie wywołania odpowiedniej procedury sprawdzającej zabezpieczenie, wystarczy znaleźć rozkaz skoku do niej i usunąć go. Jest to prostsze, niż próba tworzenia duplikatu przez ominięcie sposobu zabezpieczania. Jednak przyjęcie pewnego stylu programowania - bałaganiarskiego i niezgodnego z zasadami, których normalnie uczy się programistów, ▪ także wykorzystywanie języków wysokiego poziomu i tzw. kompilatorów optymalizujących - można skutecznie utrudnić rozszyfrowanie procedury zabezpieczającej.
Dodatkową ochroną przed usunięciem zabezpieczenia jest stosowanie wewnętrznych zegarów kontrolujących czas wykonywania kodu sprawdzającego. Programy, z których usunięto całość lub część kodu procedury zabezpieczającej, mogą być wykonywane zbyt szybko lub zbyt wolno. Ze względu na to, że zegary mogą być ustawiane w różnych fragmentach programu i ponieważ pracują one stale (również wtedy, gdy program został zatrzymany np. działaniem debbugera) - złamanie takich programów staje się bardzo trudne. Jeżeli jednak standardowy napęd dysków działa nieco szybciej lub wolniej niż powinien, mogą wystąpić trudności przy pracy z tak zabezpieczonym programem.
Omówione dotychczas zabezpieczenia miały charakter defensywny. Pojawiła się też metoda, w której użytkownik jest karany za próbę naruszenia praw autorskich. Jest ona realizowana przez użycie specjalnego programu (znanego pod nazwą „Prolok Plus" lub „Killer Prolok") dołączonego do oprogramowania, które ma zostać zabezpieczone. Sprawdza on m. in., czy któreś z przerwań sprzętowych generowanych przez sterownik dysków nie zostało „przechwycone" przez inny program (co wskazywałoby, że to nie program użytkowy sprawuje kontrolę nad sprzętem). Jeżeli taka obsługa przerwania zostanie znaleziona, to sprawdzana jest ta część kodu, która kontroluje legalność kopii. Brak dowolnego fragmentu kodu (np. zwracanie przez procedurę tej samej wartości niezależnie od wyniku jej działania) - będzie uznany za wystarczający dowód na to, że próbowano złamać zabezpieczenie. Jeśli „Killer Prolok" stwierdza, że ma do czynienia z nielegalną kopią - żąda, aby użytkownik natychmiast wyłączył komputer. W przeciwnym wypadku, po określonym czasie, zaczyna niszczyć tablicę alokacji zbiorów, co jest równoznaczne z utratą wszystkich plików z dysku, a więc i tych, które nie były obiektem zabezpieczania.
Wobec gwałtownego sprzeciwu opinii publicznej nigdy nie podjęto sprzedaży programu „Killer Prolok" na większą skalę.

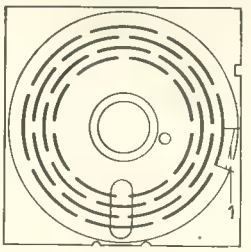


2. Metoda dodatkowej ścieżki 1-Dodatkowa ścieżka

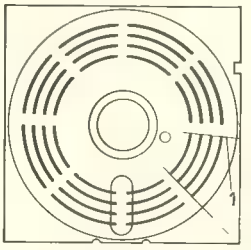




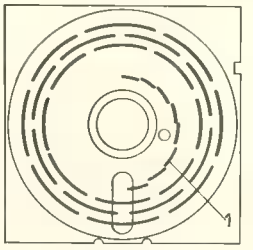
3) Metoda dodatkowych sektorów: a) umieszczenie dodatkowego sektora; b)umieszczenie nagłówków sektorów dodatkowych 1- osiem sektorów ▪ ścieżce; 2- dziewięć sektorów ▪ ścieżce; 3- nagłówki sektorów dodatkowych



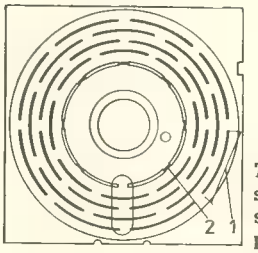
4. Metoda słabych bitów 1- obszar słabych bitów



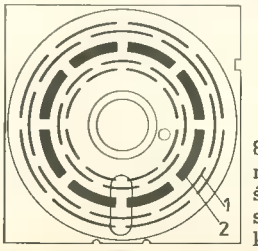
5. Metoda wyrównanych sektorów 1-linia wyrównania sektorów



6. Metoda spiralnych ścieżek 1- ścieżka spiralna



7. Metoda supersektora 1- sektor standardowy ▪ -super-sektor



8. Metoda poszerzonych ścieżek 1- ścieżka normalnej szerokości; 2- ścieżka szeroka

Zabezpieczanie oprogramowania na sztywnych dyskach

Zabezpieczanie programów zapisanych na dyskietkach ma sens dopóki dyskietka jest jedynym rodzajem pamięci masowej. Co robić, gdy użytkownik chce skopiować program na sztywny dysk?

Stosowane są dwie metody zabezpieczania pozwalające na przepisanie chronionych programów na sztywny dysk i używanie ich, ale bez możliwości kopiowania. Pierwsza wykorzystuje tzw. dyskietkę-klucz. Program (uruchamiany ze sztywnego dysku) sprawdza oryginalność zabezpieczonej dyskietki-klucza. Takie rozwiązanie ma jednak podstawowe wady: jest niewygodne, ▪ zarazem mało skuteczne, skoro program wymaga dyskietki-klucza zazwyczaj tylko raz, na początku pracy.
Coraz popularniejsza staje się inna metoda: technika programów instalujących. Użytkownik wykorzystuje firmowy program przepisujący zawartość zabezpieczonej dyskietki na sztywny dysk. Równocześnie na oryginalnej dyskietce zmieniane są pewne dane wskazujące, że program został już zainstalowany. Następnie na sztywnym dysku zapisywany jest kod, który uniemożliwia dalsze kopiowanie programu. Kod ten jest najczęściej umieszczany w ukrytym pliku i zawiera zaszyfrowany zapis informacji określających miejsce, w którym program znajduje się na dysku oraz inne dane opisujące konfigurację systemu. Szansa przekopiowania programu na inny dysk dokładnie w to samo miejsce jest bliska zeru.

Jak dotąd nie udało się opracować idealnego zabezpieczenia. Większość zachodnich producentów oprogramowania pogodziła się z faktem istnienia procederu komputerowego piractwa. Np. firma Microsoft przyjęła zupełnie inną taktykę - nie zabezpiecza swoich produktów, ale jej programy wymagają bogatej dokumentacji - ▪ zrobić odbitkę kserograficzną z podręczników (instrukcje do kompilatora języka „C" liczą ponad tysiąc stron) jest znacznie trudniej, niż skopiować w ciągu pięciu minut program zapisany na dyskietce.

Oprac. **Jan Ilnob**
wg IEEE Spectrum 6/86

Sprzętowy klucz

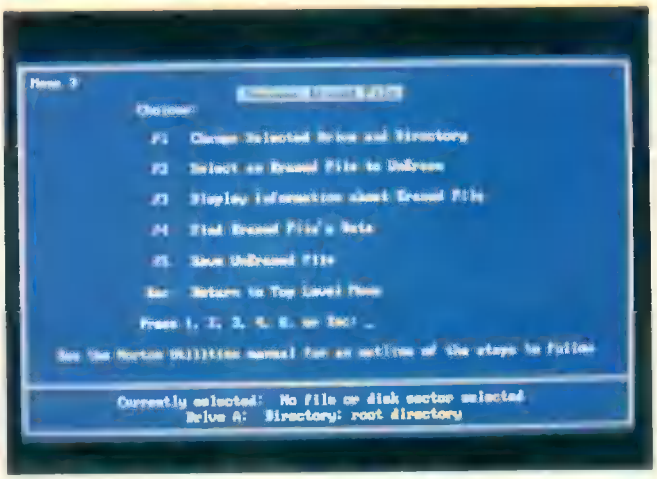
Większość firm sprzedających specjalistyczne (a co za tym idzie bardzo drogie) oprogramowanie zdecydowała się na nowy sposób ich zabezpieczania. Jest on prosty i wymaga sprzętowego „klucza" (ang. hardware key) oraz pudełka, w które ten klucz można włożyć. Kluczem jest mały układ pamięci ROM (zalany żywicą epoksydową), zawierający fragmenty kodu programu (czasami mogą być to tylko bity zmieniające pojedyncze rozkazy), bez których program na dysku jest bezużyteczny. W odróżnieniu od innych schematów zabezpieczania, sprawdzających oryginalność programu tylko podczas ładowania go do pamięci, ta metoda wymaga ciągłej obecności klucza - program odwołuje się do pamięci ROM przed każdym wejściem do istotniejszych procedur. Pudełko z „dziurką od klucza" jest układem pośredniczącym między kluczem a interfejsem szeregowym (lub równoległym) komputera.
Największą zaletą tej metody - oczywiście oprócz utrudnienia kradzieży oprogramowania - jest możliwość tworzenia dowolnej liczby kopii roboczych programu. Nie ma też kłopotów z instalacją na sztywnym dysku. Najistotniejszą wadą - konieczność wykorzystywania łącza interfejsu szeregowego, a w przypadku zmiany komputera - konieczność przenoszenia nie tylko oprogramowania, ale i klucza. Są to jednak niewielkie utrudnienia, zważywszy że programy zabezpieczane tą techniką (ACAD w 2.50, EE Designer) i tak wymagają dodatkowego sprzętu.
Złamanie nowego zabezpieczenia choć jest możliwe (na terenie Polski znajduje się już co najmniej jeden „złamany" program), wymaga jednak wielu dni, a może nawet miesięcy pracy.

(mam)

Ratujmy nasze zbiory

Każdy dysk używany pod kontrolą systemu operacyjnego MS-DOS jest podzielony logicznie na cztery obszary: programu ładującego, tablicy alokacji zbiorów (oryginał i kopia), katalogu i danych. Program formatujący rozróżnia tylko trzy obszary - sektory katalogu i danych nie są rozróżniane i wpisywane są tam bajty o wartości F6H. Obszar, w którym będą zapisywane łańcuchy alokacji zbiorów, jest wypełniany zerami (oprócz początkowych trzech lub czterech bajtów). W pierwszym sektorze zapisywany jest program ładujący.

1. Menu główne w programie The Norton Utilities



2. Menu funkcji odzyskania skasowanych zbiorów

Podczas umieszczania zbioru na dysku wypełniany jest opis w katalogu, tablica FAT oraz obszar danych. Dane są zapisywane w oddzielnych grupach - nazywanych klastrami(ang. cluster). Organizacja logiczna dysku opisana jest szczegółowo w artykule MS-DOS od środka (s.23).
System MS-DOS zapisując zbiór wypełnia kolejne, nie zajęte wcześniej klastry (od najmniejszego numeru). Jednak po dłuższym użytkowaniu dysku kolejne składające się na zbiór klastry najczęściej nie są zapisane w fizycznie następujących po sobie grupach. Może się nawet zdarzyć, że zbiór „rośnie w tył", a jego koniec może znaleźć się przed początkiem. Najbardziej przemieszane są zwykle zbiory na dyskietkach roboczych. Odzyskanie zbioru, który podlegał wielokrotnej edycji, znajdującego się na sztywnym dysku nie porządkowanym w ciągu paru miesięcy pracy, może okazać się zadaniem bardzo trudnym.
Gdy zbiór jest kasowany zleceniem systemowym DEL, dane nie są wymazywane z dyskietki, a w opisie w katalogu niszczony jest tylko jeden bajt. Zatarciu ulega natomiast łańcuch alokacji (czyli informacja o kolejnych grupach składających się na zbiór). W katalogu zapisane są dwie ważne informacje: gdzie zaczyna się łańcuch alokacji oraz jaka jest długość zbioru (obie nie są kasowane). Pierwsza z nich dodatkowo określa, w jakiej części dysku (w przybliżeniu) znajduje się zbiór. Dzięki drugiej wiadomo, kiedy należy skończyć przyłączanie klastrów do odzyskiwanego zbioru.

Jakie stosować narzędzia?

Istnieje wiele programów pomagających w odzyskaniu uszkodzonych zbiorów na dysku. Do najpopularniejszych należą: pakiet „The Norton Utilities" (wersje 1 i 3), „PC Tools" firmy Central Point Software oraz „Disk Explorer" firmy Quaid Software Limited. Czasami pomocne mogą się też okazać standardowe programy dostarczane z systemem MS-DOS: CHKDSK I RECOVER.
Zaletą pakietu The Norton Utilities jest obszerna dokumentacja oraz programy, napisane w języku Pascal i w języku maszynowym (wersje 3.00 i 3.10, a od niedawna 4.00). PC Tools jest narzędziem uniwersalnym, pozwalającym nie tylko odzyskiwać zbiory, ale i realizującym wiele funkcji systemowych. Program ten ma wielu zwolenników, jednak rozbudowane możliwości utrudniają koncentrację na określonych zagadnieniach. W wielofunkcyjnych programach znacznie częściej też można natknąć się na błędy nie zauważone przez producenta. Disk Explorer jest wygodnym Edytorem dyskowym - z jego pomocą można zmodyfikować dowolny 1 bit danych (nie można natomiast zmieniać fizycznego formatu dyskietki).

Odtwarzanie skasowanych zbiorów

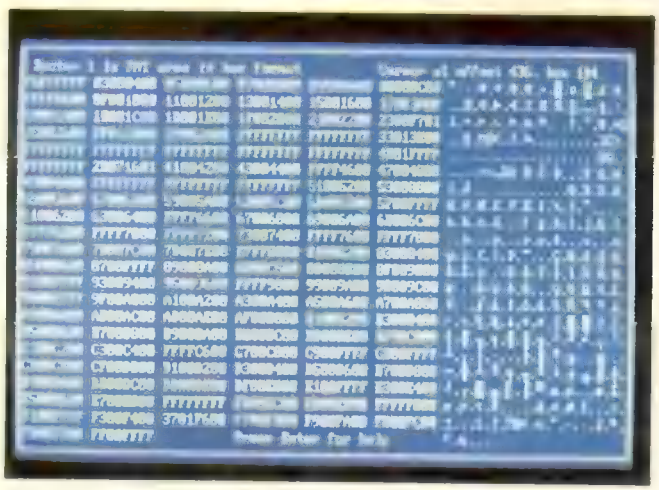
Zbiory skasowane zleceniem DEL można odzyskać posługując się programem „The Norton Utilities". W wywołaniu (zleceniem: Norton) jako parametr można podać symbol napędu dysków, na którym realizowane będą dalsze operacje. Z proponowanych przez program opcji (fot. 1) można wybrać: „odzyskanie skasowanego zbioru" (f3 Recover Erased File). Następnie ukaże się menu (fot.2), z którego należy wybrać opcję „wybierz skasowany zbiór do odzyskania" (f2 - Select an Erased File to UnErase). Ukażą się wtedy nazwy skasowanych zbiorów ze znakiem zapytania w miejscu pierwszej litery - podczas kasowania wpisywana jest tam wartość E5H. W pierwszym kroku odzyskiwania należy na nowo wpisać w to miejsce odpowiedni znak. Spowoduje to wyjście z opcji i można wtedy obejrzeć, jakie dane zawiera opis zbioru w katalogu (f3 - „wyświetl informację o skasowanym zbiorze"). Dodatkowo pojawi się informacja, czy pierwsza grupa danych początkowo zajmowana przez odzyskiwany zbiór nie została zajęta przez inny.
Następnie należy rozpocząć wyszukiwanie kolej-

nych klastrów i przyłączanie ich do łańcucji. Służy do tego opcja f4 - „znajdź dane skasowanego zbioru". Posługiwanie się funkcjami znajdującymi się w tym menu stanowi najtrudniejszy element w algorytmie odzyskiwania zbioru. Są tu następujące możliwości wyboru:
f1 automatyczne wyszukiwanie wszystkich danych,
f2 automatyczne wyszukiwanie pojedynczego klastra,
f3 poszukiwanie specyficznych danych,
f4 wybranie pojedynczej grupy według wskazanego numeru sektora.
Pierwszej opcji można użyć tylko na nowo nagranej dyskietce (zakłada się, że grupy ułożone są sekwencyjnie). Druga, bardziej skomplikowana opcja, pozwala odtworzyć zbiór, którego poszczególne klastry znajdują się w bliskim sąsiedztwie pierwszego. Po wyborze każdego klastra można

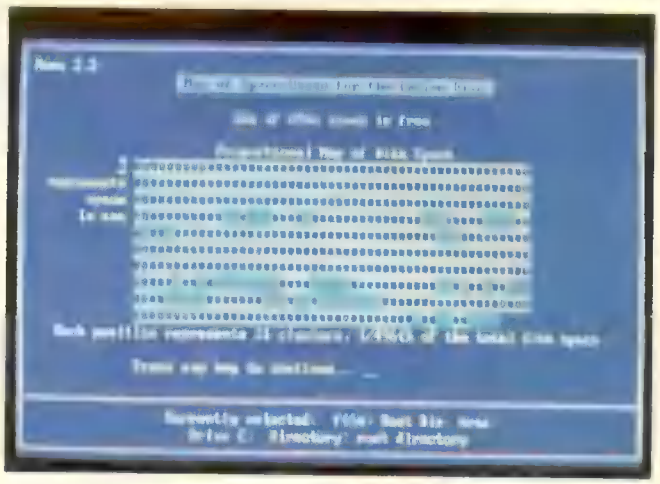
np. podczas używania programu Turbo Pascal). Szczęśliwym zbiegiem okoliczności może zachować się kopia FAT, znajdująca się zaraz za oryginałem, wystarczy wtedy ją przepisać w miejsce oryginalnej. Należy wtedy albo „na piechotę" przepisać zawartość kopii (korzystając z programu Norton), albo napisać program kopiujący sektor po sektorze. W przypadku pisania takiego programu w języku wyższego poziomu, do czytania i zapisywania danych na dysk należy wykorzystywać procedurę BIOS-a 13H (obsługa przerwań z poziomu DOS-a - 25H i 26H zachowuje na stosie zawartość rejestru znaczników).
Najczęściej jednak zostają zamazane obydwa obszary FAT. Należy wówczas zapełnić zerami obszar FAT (oprócz początkowych dwu pól) i odzyskiwać zbiory poprzednio opisywaną metodą.
Zmiany zawartości FAT można dokonać stosując

Odtwarzanie katalogów

W katalogu umieszczone są istotne dane dotyczące zbiorów znajdujących się na dysku: etykieta (zawierająca nazwę, jej rozszerzenie, atrybuty, datę i czas założenia - lub ostatniej edycji - zbioru) oraz początek łańcucha alokacji i długość zbioru. Do odczytania danych z dyskietki wystarczy, aby trzy z tych pól były poprawne: nazwa, rozszerzenie i pole określające numer pierwszej grupy sektorów.
Przy odtwarzaniu katalogu, podobnie jak poprzednio, można posłużyć się programem do edycji zapisów na dysku. Zamazaną nazwę zbioru należy wpisać „dużymi" literami (osiem znaków - jeżeli jest ich mniej, trzeba dopełnić nazwę spacjami). Trzyznakowe rozszerzenie wpisuje się zaraz za nazwą (nie należy wstawiać kropki) na podobnych zasadach, jak nazwę. Ponieważ na opis zbioru



3. 16-bitowa tablica FAT



4.Rozmieszczenie zapisów na dysku sztywnym po dłuższym czasie użytkowania

obejrzeć jego zawartość i podjąć decyzję o przyłączeniu do zbioru.
Znacznie trudniej jest odzyskiwać zbiory na długo użytkowanym sztywnym dysku - odległości pomiędzy kolejnymi klastrami mogą okazać się znaczne i czasem konieczne jest przeszukanie całego dysku. W tym przypadku bardzo przydatna jest opcja f3, szczególnie gdy odzyskiwane są pliki typu ASCII - w następnym klastrze należy szukać dokończenia wyrazu lub charakterystycznego zestawu znaków (np. instrukcja programu lub nazwa w przypadku zbioru rekordów bazy danych). Opcja f4 jest bardzo rzadko wykorzystywana.
Po zebraniu wszystkich klastrów w łańcuch, należy wrócić do menu funkcji „odzyskaj skasowany zbiór" i wybrać opcję - „zapisz odzyskany zbiór".

Naprawianie tablicy FAT

Przed rozpoczęciem modyfikacji tablicy FAT należy zapoznać się ze strukturą zawartej tam informacji (patrz „MS-DOS od środka" część 5). Modyfikacje są konieczne, gdy w obszarze FAT zostanie zapisana błędna informacja (może się to zdarzyć

np. „The Norton Utilities". W tym celu należy wybrać opcję f2 - „zmiana lub przeglądanie informacji" zawartych na dysku. Następnie w nowym menu opcją f1 - „zmień wybór", należy podać numer sektora do edycji (pierwszy sektor FAT - to sektor nr 1). Opcją f5 - „wyświetl lub zmieniaj zawartość wybranego obszaru" przechodzi się do edycji sektora i można zacząć modyfikację zapisanych tam danych. Po wprowadzeniu modyfikacji program żąda decyzji, czy zapisywać zmiany na dysk (opcja f1 - zapisz zmienione dane na dysk). Więcej kłopotu sprawia sytuacja, gdy obszary nie dość, że uległy zamazaniu, to jeszcze została wpisana zła suma kontrolna (patrz „Zabezpieczanie programów" (s. 19). Takich sektorów „The Norton Utilities" nie przeczyta (a więc i nie zapisze zmian). Można natomiast posłużyć się programem „Disk Explorer". Błędna suma kontrolna nie stanowi dla niego przeszkody - po odczytaniu takiego sektora wystarczy go zapisać, a suma kontrolna zostanie również zapisana poprawnie. Dalsze postępowanie jest analogiczne, jak dla „normalnego" uszkodzenia zapisu w obszarze FAT.

w katalogu zarezerwowane są 32 bajty, to nazwy oddalone są od siebie o tę liczbę bajtów. Pierwsza etykieta zbioru zaczyna się zaraz na początku obszaru katalogu.
Największy kłopot jest z pierwszym ogniwem łańcucha alokacji: trzeba znaleźć grupę, od której zaczyna się zbiór. Następnie należy określić jej numer według algorytmu podanego w artykule „MS DOS od środka" cz. 5 (s...). Dopiero tę wartość możemy wpisać w 26 i 27 bajt pola opisującego zbiór w katalogu. Tak przygotowany zapis w katalogu zwykle wystarczy do odczytania zbiorów z poziomu MS-DOSa.

Uwaga: Wszelkie próby reperowania dyskietki należy wykonywać na ich kopiach, zrobionych za pomocą odpowiedniego programu kopiującego (np. Copywrite lub CopyIIPC). Pozwala to uniknąć konsekwencji ewentualnych błędów. W przypadku dysków sztywnych utworzenie takiej kopii jest niemożliwe, ale można spróbować podobnie uszkodzić zawartość dyskietki i dopiero na takim modelu planować naprawę.

Maciej A. Markowski

UŁATWIAMY ZARZĄDZANIE
STUDIO USŁUG KOMPUTEROWYCH sp. z o. o.
Samba
BIURO HANDLOWE:
ul. Władysława IV
81-384 Gdynia
☎ 21 70 88.

(5) MS-DOS OD ŚRODKA

W poprzedniej części omawiane były uniwersalne wywołania funkcyjne. Oto pozostałe.

Funkcja 27H umożliwia swobodny odczyt jednego lub wielu rekordów ze zbioru dyskowego i zapisanie ich do obszaru DTA. Rejestry DS:DX wskazują na blok FCB, w którym podana jest nazwa zbioru (przemieszczenie 01H), jej rozszerzenie (przemieszczenie 08H) oraz numer pierwszego rekordu (przemieszczenie 21H do 24H). Rejestr CX zawiera liczbę odczytywanych rekordów i musi być różny od 0. Po powrocie z wywołania funkcji rejestr AL zawiera kod błędu w takim samym formacie jak dla funkcji 14H lub 21H (odczyt sekwencyjny lub swobodny jednego rekordu), zaś CX - liczbę faktycznie przeczytanych rekordów. W przeciwieństwie do funkcji 21H, uaktualniany jest numer rekordu przy dostępie swobodnym (przemieszczenie 21H...24H) - przyjmowany jest numer rekordu następnego po ostatnim odczytanym. Jednocześnie uaktualniane są wskaźniki wykorzystywane przy dostępie sekwencyjnym: numer bloku (przemieszczenie 0CH...0DH) i numer rekordu w ramach bloku (przemieszczenie 20H).

Funkcja 28H umożliwia swobodny zapis jednego lub wielu rekordów do zbioru dyskowego. Nazwa zbioru i jej rozszerzenie oraz numer pierwszego rekordu podawane są w bloku FCB, wskazywanym przez rejestry DS:DX, zaś liczba rekordów - w rejestrze CX. Umieszczenie wartości 0 w rejestrze CX sygnalizuje, że przetwarzany zbiór ma się kończyć przed wyspecyfikowanym rekordem. Umożliwia to w prosty sposób zwalnianie miejsca na dysku, gdy końcowa część zbioru przestaje być potrzebna.

Po powrocie z wywołania funkcji rejestr AL zawiera kod błędu o takim samym znaczeniu, jak przy funkcji 15H lub 22H (zapis sekwencyjny lub swobodny jednego rekordu), zaś CX - faktyczną liczbę zapisanych rekordów. Uaktualniane są: numer rekordu przy dostępie swobodnym, numer bloku i numer rekordu w ramach bloku przy dostępie sekwencyjnym.

Uwaga: po otwarciu zbioru (za pomocą funkcji 0FH) w pole FCB o przemieszczeniu 0EH i 0FH otwieranego zbioru wpisywana jest wielkość rekordu wynosząca 128 bajtów. Wpisując w to pole inną wartość, można zmieniać rozmiar rekordu. Często wpisuje się wartość 1 i wówczas przy każdym odczycie lub zapisie rekordu przesyłany jest jeden bajt. Obszar transmisji dyskowych (DTA) można zmieniać za pomocą funkcji 1AH.

Funkcja 29H analizuje wprowadzony z urządzenia wejściowego wiersz polecenia, poszukując specyfikacji zbiorów o postaci DRIVE:FILENAME.EXT, gdzie DRIVE jest symbolem napędu dysków, zaś FILENAME i EXT są nazwą zbioru i jej rozszerzeniem. Funkcja nie pozwala na podawanie ścieżki dostępu (ang. pathname). Rejestry DS:SI wskazują obszar pamięci, w którym znajduje się analizowany ciąg znaków, rejestry ES:DI - obszar pamięci, w którym w razie znalezienia zbioru system tworzy nie otwarty blok FCB. Bity 0...3 rejestru AL określają sposób analizowania ciągu znaków:

- bit 0 równy 1 sprawia, że pomijane są początkowe separatory (tabulacja, spacja, plus i przecinek), bit 0 równy 0 oznacza, że specyfikacja pliku rozpoczyna się bezpośrednio pod adresem DS:SI;
- bit 1 równy 1 sprawia, że bajt zawierający numer napędu (przemieszczenie 00H) będzie zmieniony tylko wówczas, gdy numer napędu został podany w specyfikacji zbioru. Dzięki temu blok FCB może mieć wpisany przez programistę własny domyślny numer napędu, inny niż systemowy;
- bit 2 równy 1 sprawia, że blok FCB jest zmieniany tylko w razie podania prawidłowo skonstruowanej nazwy zbioru. Dzięki temu w programach można używać domyślnych nazw zbiorów, które mogą być zastąpione innymi bezpośrednio podanymi nazwami;
- bit 3 równy 1 sprawia, że blok FCB jest zmieniany tylko w razie podania prawidłowo skonstruowanego rozszerzenia nazwy zbioru.

Symbol wieloznaczny „*" zastępowany jest wynikającą z kontekstu liczbą symboli wieloznacznych „?". Po powrocie z wywołania funkcji zawartość rejestru AL świadczy o wyniku analizy:

AL=0 oznacza znalezienie jednoznacznie określonego zbioru;
AL=1 oznacza znalezienie zbioru określonego za pomocą nazwy wieloznacznej;
AL=0FFH świadczy, że nie znaleziono zbioru o podanej nazwie lub źle go wyspecyfikowano.

Wartość rejestrów DS:SI (praktycznie tylko rejestr SI) jest zmieniana w ten sposób, że wskazuje na pierwszy bajt za przeanalizowanym ciągiem znaków. Umożliwia to ewentualne bezpośrednie przejście do analizy kolejnej specyfikacji zbioru.

Funkcja 2FH przekazuje w rejestrach ES:BX adres aktualnego obszaru transmisji dyskowych (DTA). Została ona wprowadzona dopiero w wersji 2.0 MS-DOS, jednak dla użytkowników funkcji rozszerzonych nie ma większego znaczenia (gdyż nie odwołują się oni do DTA).

Tabela 1. Liczba sektorów w poszczególnych polach dla standardowych formatów dyskietek.

Średnica dyskietki:	5,25"				8"				
Gęstość zapisu:	podwójna				poczwórna	pojedyncza	pojedyncza	podwójna	
pole zarezerwowane	1	1	1	1	1	1	1	4	1
pierwsza kopia FAT	1	3	1	3	5	7	5	6	2
druga kopia FAT	1	3	1	2	5	7	5	6	2
katalog główny	4	4	7	7	7	14	17	17	6
pole zbiorów	313	351	630	708	1422	2371	1972	1929	605

Tabela 2. Parametry standardowych formatów dyskietek

Średnica dyskietki:	5,25"				8"				
Gęstość zapisu:	podwójna				poczwórna	pojedyncza	pojedyncza	podwójna	
liczba sektorów na ścieżce	8	9	8	9	8	15	26	26	8
liczba bajtów w sektorze	512	512	512	512	512	512	128	128	1024
liczba stron	1	1	2	2	2	2	1	1	2
liczba ścieżek na stronie	40	40	40	40	80	80	77	77	77
liczba sektorów w jednostce alokacji	1	1	2	2	2	1	4	4	1
całkowita liczba sektorów	320	360	640	720	1440	2400	2002	2002	616
nominalna pojemność	160K	180K	320K	360K	720K	1200K	250.25K	250.25K	1232K
liczba pozycji w katalogu głównym	64	64	112	112	112	224	68	68	192
deskryptor nośnika	FE	FC	FF	FD	F9	F9	FE	FD	FF

Organizacja logiczna dysków

Większą część dysku zajmują dane zapisane w zbiorach, niezbędny jest jednak pewien obszar na informacje organizujące dane w spójną całość. Dysk podzielony jest logicznie na kilka pól, a liczba sektorów w każdym z pól zależy od formatu dyskietki (tab. 1). Istnieją też inne, nie opisane w tabeli formaty (w szczególności formaty dla dysków sztywnych). Można je stosować po przyłączeniu odpowiedniego programu sterującego (ang. device driver).
Informacja wchodząca w skład zbioru dyskowego umieszczana jest w polu danych poszczególnych sektorów. Dla zredukowania rozmiarów tablicy, w której rejestrowane są informacje o położeniu kolejnych fragmentów zbioru postanowiono stworzyć grupy sektorów czyli tzw. jednostki alokacji (ang. cluster). Klaster* może się składać z jednego lub z kilku sektorów - zależy to od formatu dysku (tab. 2). Zbiór nie może więc zajmować na dysku mniej miejsca niż zajmuje jednostka alokacji.

FAT - tablica alokacji zbiorów

Informacja o klastrach składających się na poszczególne zbiory (lub jeszcze nie zapisanych) przechowywana jest w FAT (ang. file allocation table - tablica alokacji zbiorów). Dla większego bezpieczeństwa danych na dyskietce przechowuje się co najmniej dwie identyczne tablice alokacji.

Tabela 3. Znaczenie wartości zapisywanych w FAT

Zawartość pozycji w FAT	Status klastra w polu dysku przeznaczonym dla zbiorów
000	klaster nie jest zapisany
FF7	klaster zawiera uszkodzony sektor
FF8H...FFFH	ostatni klaster w zbiorze dyskowym
XXX	każda inna wartość jest numerem pozycji FAT odpowiadającej następnemu klastrowi zbioru

Tabela 4. Możliwe funkcje pozycji katalogu dyskowego

Kody dla 2 pocz. bajtów (HEX)	Funkcja pozycji katalogu
2E 2E	Kody 2E (znak kropki w kodzie ASCII) oznaczają odwołanie do katalogu nadrzędnego. W tym wypadku bajty 1A przemieszczeniu 1AH...1BH względem początku danej pozycji katalogu zawierają numer pozycji w FAT odpowiadającej pierwszemu klastrowi zawierającemu katalog nadrzędny (000 jeśli jest to katalog główny).
2E 20	Kropka i spacja oznaczają odwołanie do bieżącego katalogu (czyli zawierającego tę pozycję). Bajty 1AH...1BH wskazują na pierwszą grupę sektorów zawierających ten katalog.
00 xx	Dana pozycja nie była jeszcze nigdy używana
E5 xx	Pozycja była używana, ale zbiór został skasowany.
inny kod (znak ASCII)	Pierwszy znak nazwy. Pozycja jest odwołaniem do zbioru (który w szczególności może być katalogiem podrzędnym) albo zawiera etykietę nośnika (tylko w przypadku katalogu głównego). Do dalszego rozróżnienia (plik czy etykieta nośnika) służy atrybut pliku (bajt o przemieszczeniu 0BH).

Pole tablicy FAT podzielone jest na szereg 12-bitowych pozycji, numerowanych od 0. Dwie początkowe pozycje (0 i 1), zajmują w sumie 24 bity, czyli 3 bajty. Pierwszy bajt to tzw. deskryptor formatu nośnika (tab. 2), umożliwiający identyfikację formatu stosowanej dyskietki, dwa dalsze - są równe FFH.
Pozostałe 12-bitowe pozycje FAT odpowiadają poszczególnym jednostkom alokacji dla pola przeznaczonego na zbiory: pozycja 2 - pierwszemu klastrowi, pozycja 3 - drugiemu itd. 12-bitową zawartość każdej pozycji można wyrazić za pomocą trzech cyfr szesnastkowych. Jeśli dany klaster w polu zbiorów nie jest zapisany, to odpowiadająca mu pozycja FAT ma wartość 000. Po zapisaniu klastra odpowiadająca mu pozycja w FAT zmienia wartość (tab. 3).
Wartości zapisywane w kolejnych lokacjach FAT umożliwiają nie tylko rozróżnienie, które klastry są zajęte, a które wolne, lecz także pozwalają na odnalezienie kolejnych grup składających się w zbiór. W każdym katalogu wśród informacji opisujących dany zbiór znajduje się m. in. numer pozycji w FAT, odpowiadającej pierwszemu klastrowi zbioru. Pozycja ta zawiera numer pozycji opisującej następny klaster zbioru lub wartość FF8H...FFFH, gdy jest to ostatni klaster (tab. 3). Dla zbiorów składających się z wielu klastrów powstaje struktura listowa, zaczynająca się w skorowidzu i przebiegająca przez pozycje FAT odpowiadające kolejnym klastrom należącym do zbioru.

Organizacja katalogów

Katalog główny nie jest zbiorem dyskowym, lecz zajmuje określone miejsce na dysku i ma stałą długość, zależną od formatu dysku (tab. 1). Katalogi podrzędne są zbiorami i mają podobną budowę jak katalog główny, ale ich długość nie jest ograniczona, gdyż (w przeciwieństwie do katalogu głównego) mogą zajmować wiele klastrów. Katalog główny składa się więc z określonej liczby pozycji (tab. 2), natomiast w katalogu podrzędnym liczba pozycji jest ograniczona tylko przez pojemność nośnika. O rodzaju informacji zawartej w danej pozycji katalogu decyduje jej pierwszy (i ewentualnie drugi) bajt (tab. 4).
Opis poszczególnych pól występujących w każdej pozycji katalogu przedstawiono w tabeli 5.

Wyszukiwanie sektorów należących do zbioru dyskowego

Przez odczytanie deskryptora nośnika można określić format dysku. Jest to konieczne przed rozpoczęciem wyszukiwania sektorów logicznych należących do danego zbioru. Deskryptor nośnika jest pierwszym bajtem w FAT, czyli pierwszym bajtem za obszarem zarezerwowanym (na dyskietkach 5,25" pierwszym bajtem sektora 1). Można go odczytać posługując się programem DEBUG lub za pomocą funkcji systemowych 1BH lub 1CH.
Znając format dysku i jego parametry (tab. 1 i 2), można określić numery sektorów zawierających tablicę FAT oraz numer pierwszego sektora pola przeznaczonego dla zbiorów. Teraz, z pozycji katalogu opisującej dany zbiór, można odczytać numer pozycji FAT odpowiadającej pierwszej grupie sektorów składających się na zbiór. Następnie trzeba zlokalizować pozycje w FAT, odpowiadające kolejnym klastrom zbioru. Należy w tym celu wykonać czynności:
- numer zapisany w bieżącej pozycji tablicy FAT pomnożyć przez 1,5 (każda pozycja ma 12 bitów, czyli zajmuje półtora bajta);
- część całkowita wyniku jest przemieszczeniem w tablicy FAT określającym położenie 2-bajtowego słowa zawierającego informację o następnej grupie sektorów;
- jeśli bieżąca pozycja w FAT ma numer nieparzysty, należy wziąć 12 mniej znaczących bitów wyszukanego słowa; w przeciwnym wypadku 12 bardziej znaczących bitów;
- jeśli otrzymana w ten sposób liczba jest równa FF8H...FFFH, zbiór nie zawiera więcej klastrów. W przeciwnym razie liczba ta jest numerem pozycji w FAT odpowiadającej następnej grupie sektorów zbioru.

Aby na podstawie numeru pozycji w FAT określić numery logiczne sektorów wchodzących w skład klastra, należy:
- odjąć 2 od numeru pozycji FAT (bo pozycja w FAT odpowiadająca pierwszemu klastrowi ma numer równy 2);
- pomnożyć wynik przez liczbę sektorów składającą się w danym formacie na grupę;
- dodać do wyniku numer sektora, od którego rozpoczyna się obszar przeznaczony na zapis zbiorów.

Uzyskuje się w ten sposób numer pierwszego z sektorów wchodzących w skład danego klastra.
Uwaga: Jeśli format dysku nie jest znany, to:
- funkcje 1BH i 1CH pozwalają odczytać deskryptor nośnika i położenie początku FAT;
- funkcja 36H pozwala określić długość FAT (zawartość rejestru DX pomnożona przez 1,5);
- początek pola przeznaczonego dla zbiorów można znaleźć zapisując na czysty dysk znany zbiór i sprawdzając, gdzie został zapisany.

Opr. Zbigniew Pojmański

wg mc 2/1986 i Norton P. „Programer's Guide to the IBM PC".

* Dokładne tłumaczenie angielskiego słowa cluster (grono lub kiść) nie nadaje się do wykorzystania w polskich tekstach. Inne słowa, takie jak: zlepek, porcja lub grupa nie są oczywistymi odpowiednikami, tzn. nie każdy zorientuje się, że chodzi o „cluster". Podobny problem mieli już fizycy zajmujący się astronomią i ... wprowadzili do swojego języka spolszczenie klaster.

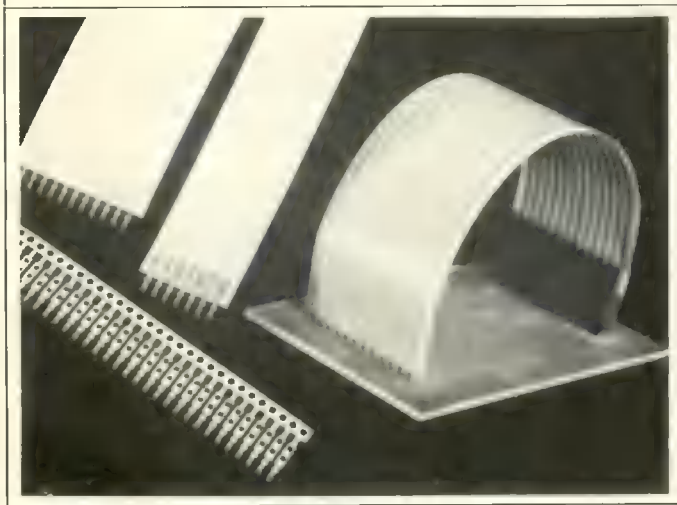
Pola FAT o rozmiarach 12-bitów umożliwiają na dysku 4080 grup (jednostek alokacji). Wystarczyło to dla IBM PC XT ze sztywnym dyskiem 10 MB. Z chwilą gdy popularne stały się sztywne dyski o większej pojemności wprowadzono nowy 16-bitowy format dla pozycji w tablicy FAT. Firma IBM zrealizowała to w modelu PC AT, który wyposażony został standardowo w sztywny dysk 20 MB. Format dwubajtowy jest logicznym rozszerzeniem formatu półtorabajtowego. Specjalne wartości pól nie różnią się między sobą tylko dodaniem pierwszej cyfry FH w formacie dwubajtowym - np. koniec pliku jest oznaczony FFFFH (zamiast FFF), natomiast znacznik uszkodzenia sektora w grupie FFF7H (zamiast FF7H). Dodatkową zaletą formatu 16-bitowego jest większa przejrzystość tablicy i możliwość uproszczenia programu obsługi. Przyspiesza to wyszukiwanie poszczególnych grup należących do pliku.
Początkowe dwa pola w FAT nie mają zwykłego znaczenia: pierwsze z nich określa typ dysku, drugie ma zawsze wartość FFFH (pole 12-bitowe) lub FFFFH (pole 16-bitowe). Następne pola mogą przybierać wartość 3...FF0H (lub FFF0H), określając położenie następnego z pól w łańcuchu lub wartość FFF czy FF7. Wartości FF8...FFE są zarezerwowane do wykorzystania w przyszłości. Podczas zapisywania na dysk liczb większych od FFH następuje odwrócenie kolejności bajtów (jako pierwszy zapisywany jest mniej znaczący bajt). Dotyczy to pól w FAT i opisów pliku w katalogu.

(mam)

mikrokram

Łączniki

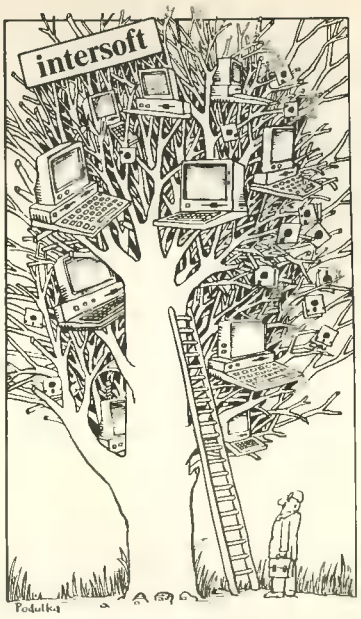
Nowy system łączników do elastycznych połączeń obwodów drukowanych, przełączników i innych urządzeń elektrycznych stosowanych w sprzęcie radiowym i TV oraz - coraz częściej - w samochodowych deskach rozdzielczych opracowano w firmie Du Pont. System przeznaczony jest specjalnie do bezpośredniego lutowania z obwodami drukowanymi i układami scalonymi. Wyprowadzenia długości 2,3 lub 4,3 mm łączone są z przewodami na prasie ręcznej (2 do 24 wyprowadzeń w ciągu 6...8 s) lub pneumatycznej (36 wyprowadzen w ciągu 4...6 s) w sposób gwarantujący dobre połączenie elektryczne i odporność na wstrząsy. Zestawy końcówek dostarczane są w oddzielnych taśmach lub rolach. DU PONT Information Service **(pc)**



Zachód łagodzi kontrolę

We wrześniu na posiedzeniu w Paryżu CoCom (Coordinating Comittete for Multinational Export Controls) ma zamiar zmniejszyć ograniczenia w eksporcie komputerów osobistych z Zachodu do „krajów Bloku Wschodniego". Z listy artykułów objętych ścisłym zakazem eksportu do krajów socjalistycznych mają zostać skreślone komputery 8-bitowe do klasy Apple II włącznie. Producenci i handlowcy oczekują, że CoCom zezwoli na sprzedaż IBM PC XT a nawet IBM PC AT, jako że powszechna dostępność tego sprzętu w sklepach czyni wszelkie zakazy absurdalnymi.

Zmniejszenie ograniczeń eksportowych nie oznacza, że cały sprzęt w klasie „komputer domowo-biurowy" zostanie skreślony z długiej listy artykułów objętych embargiem. Specjaliści CoCom zwracają uwagę, że nawet 8-bitowy mikroprocesor może być użyty do celów militarnych.
Financial Times **(zh)**



PROWADZI
KOMPLEKSOWĄ OBSŁUGĘ PRZEDSIĘBIORSTW
W ZAKRESIE KOMPUTERYZACJI

w tym:

● instalację sieci komputerowych ● instalację programów specjalistycznych (księgowość, gospodarka magazynowa i materiałowa, kosztorysowanie, gospodarka transportowa, optymalizacja, planowanie, statystyka).

Posiadamy bogaty wybor literatury i oprogramowania w języku polskim (AutoCAD, Lotus 1-2-3, Symphony, Framework, DOS 3.2, Turbo-Pascal, Fortran 77)

Prowadzimy sprzedaż sprzętu komputerowego.

intersoft

sp. z o.o. 00-160 Warszawa, ul. Zamenhofa 4 m.32 ☎ 31-63-22

eurobit

Al.Ujazdowskie 18 m. 14
00-478 Warszawa
☎ 28-01-76

Sprzęt komputerowy :
Komputery
Drukarki
Stacje dysków
Koprocesory

Opisy programów
Instrukcje obsługi komputerów
Mapy pamięci
Opisy języków programowania

Bogate oprogramowanie IBM, ATARI ST AMSTRAD.....
Szybka realizacja programów pisanych na zamówienie

Płytka z

Wspomaganie projektowania obwodów drukowanych

Pierwszymi urządzeniami wspomagającymi projektowanie obwodów drukowanych były tzw. digitajzery, które pojawiły się w Polsce we wczesnych latach siedemdziesiątych. Składały się one z komputera z zewnętrzną pamięcią taśmową i specjalnego stołu o podłożu stabilnym termicznie, służącego do elektronicznego odwzorowywania rysunków. Ręcznie wykonany projekt w skali 2:1 (lub 4:1 przy większych gęstościach upakowania) wprowadzano do pamięci operacyjnej komputera, a następnie zapisywano w pamięci taśmowej. Informacje zgromadzone na taśmie służyły do sterowania urządzeń wyjściowych, które tworzyły dokumentację produkcyjną (klisze poszczególnych warstw ścieżek, taśmy sterujące wiertarką itp.).
Ostatnio wraz ze wzrostem liczby komputerów pojawiły się w Polsce programy do projektowania obwodów drukowanych. Często zbyt proste (nie pozwalające na przesłanie projektu do odpowiednich urządzeń wyjściowych), by możliwe było uzyskanie niezbędnej dokumentacji produkcyjnej. Istnieje jednak w kraju kilka profesjonalnych systemów wspomagających projektowanie, w których zastosowano specjalizowany sprzęt i oprogramowanie. Jednym z nich jest CAD SYSTEM 80/10, wyprodukowany przez szwajcarską firmę EIE (Electronic Industrial Equipment SA) wykorzystywany w Zakładach Wytwórczych Urządzeń Telefonicznych w Warszawie.
Jednostką centralną systemu jest procesor PDP 11/23 z pamięcią operacyjną 256 KB. Pamięć masowa to dwie stacje dysków elastycznych 5,25" oraz dysk sztywny 10 MB. Do systemu przyłączone są specjalizowane urządzenia peryferyjne - digitajzery, ploter i fotoploter (o rozdzielczości 0,01 mm) oraz urządzenia pomocnicze - monitor kolorowy o dużej rozdzielczości, drukarka i perforator. System można rozbudować do siedmiu stanowisk projektowych. Można też wykorzystać konwerter CAM-OFF, zezwalający na wymianę zadań projektowych między standardami RX50, GERBER i PC/MS-DOS, przyłączenie dodatkowych urządzeń peryferyjnych oraz wykonywanie modyfikacji projektów (co odciąża CAD 80/10).
Oprogramowanie systemowe (5 MB) przechowywane jest na dysku sztywnym. System ma własny program zarządzający, korzystający tylko z firmowej procedury wykonawczej ODT. Pozostałe 5 MB sztywnego dysku jest wykorzystywane przez projektanta obwodu drukowanego. Projekty o średniej gęstości upakowania, np. 30 układów scalonych w obudowach typu DIL16 na powierzchni 3,2 dm^2 (co odpowiada typowej karcie IBM PC XT) zajmują 80 KB -przy metodzie projektowej z odwzorowaniem schematu ideowego.
System posiada bibliotekę elementów elektronicznych (np. seria 74, Intel, Zilog) oraz bibliotekę obudów (np. RES8 dla oporników 0,25 W, CAPE4 dla kondensatorów elekrolitycznych, DIL16 dla układów scalonych itd.). Można je modyfikować i rozszerzać za pomocą odpowiednich programów.
Oprogramowanie CAD 80/10 umożliwia rozpoczęcie prac na jednym z następujących poziomów:
- odwzorowanie schematu ideowego;
- wprowadzenie wcześniej przygotowanej listy elementów z typami obudów i rozmiarami punktów lutowniczych oraz listy połączeń;
- digitalizacja projektu wykonanego tradycyjną metodą na podłożu z siatką metryczną lub calową.

Odwzorowanie projektu według schematu ideowego polega na podaniu elementów (spośród zapamiętanych w bibliotece systemu) i połączeniu ich zgodnie ze schematem. Schemat ideowy jest podstawą do automatycznego utworzenia listy elementów, połączeń i nie wykorzystanych wyprowadzeń. Programy kontrolne mogą na tym etapie wykryć powtarzające się lub błędne połączenia (np. zwarcia). Następnie wprowadzane są dane o wymiarach płytki.
Firma EIE opracowała program do automatycznego rozmieszczania elementów, ale okazało się, że nawet średnio doświadczony operator może uzyskiwać lepsze wyniki. Jeśli natomiast rozmieszczenie elementów nie jest optymalne, to zdarza się, że dla

Smartwork

Program SMARTWORK, opracowany przez amerykańską firmę WINTEK, umożliwia konwersacyjne wykonywanie projektów płytek o rozmiarach nie przekraczających 10"x16" (254x406,4 mm) jedno- i dwuwarstwowych przy wykorzystaniu komputera osobistego IBM (lub kompatybilnego). Program wymaga sprzętu o następującej konfiguracji:
- minimum 192 KB pamięci operacyjnej RAM
- jeden napęd dysków (w starszej wersji dwa)
- monitor monochromatyczny lub kolorowy (przy projektowaniu druku dwuwarstwowego)
- ploter lub drukarka graficzna
- opcjonalnie mysz.

Projektowanie przy użyciu programu SMARTWORK pozwala przygotować dokumentację do wykonywania płytek drukowanych, przy częściowym zastosowaniu tradycyjnej techniki (aparat fotograficzny do przenoszenia projektu druku na płytkę). Możliwe jest też szybkie modyfikowanie istniejącego projektu oraz archiwizacja prac.
Na dyskiecie z programem SMARTWORK znajdują się cztery pliki:
EDIT.EXE
PLOT.EXE
SMART.STA (w nowej wersji SMART.EXE, PLOT.EXE)
DEMO.PCB
Pierwszy z nich - EDIT.EXE jest programem głównym, umożliwiającym wyświetlenie projektu na ekranie monitora oraz prowadzenie prac edytorskich na wykonanej części projektu.
PLOT.EXE program obsługi urządzeń rysujących (ploter, drukarka).
W starszej wersji: SMART.STA jest zbiorem danych dla programów typu .EXE, a DEMO.PCB jest plikiem demonstracyjnym.
Program SMARTWORK umożliwia korzystanie z następujących elementów druku:
ścieżka cienka 0,012" (0,3 mm)
ścieżka gruba 0,050" (1,27 mm)
punkt lutowniczy o średnicy 0,062" (1,6 mm)
minimalny odstęp 0,019" (0,48 mm)
Punkty lutownicze są w kształcie kół (dla zachowania minimalnego dopuszczalnego odstępu punkt lutowniczy przybiera czasem kształt koła ściętego cięciwą równoległą do sąsiadującej z nim ścieżki). SMARTWORK umożliwia wprowadzenie pojedynczych punktów, rzędu lub dwu rzędów punktów (moduł układu scalonego). Najmniejsza odległość punkt-punkt lutowniczy wynosi 0,050". W razie potrzeby można dokonywać uzupełnień w zaprojektowanym obwodzie drukowanym, powiększać powierzchnię lutowniczą wokół otworu, itp.
Projektowanie połączeń obwodu drukowanego przy użyciu SMARTWORK-a odbywa się automatycznie. Ścieżki między wskazanymi za pomocą kursora punktami lutowniczymi i na wybranej przez projektanta warstwie są prowadzone po najkrótszej drodze (pionowo, poziomo lub pod kątem 45°) z zachowaniem odpowiedniej szerokości ścieżek i odległości między nimi. Istnieje możliwość zmiany ich szerokości, skasowania, połączenia, uzupełnienia o nowe oraz kilkukrotnego powtórzenia wyznaczonej ścieżki.
Można również uzyskać ścieżki o grubości będącej wielokrotnością wymiaru 0,050".
Moduł jednostronnego punktu lutowniczego można wykorzystać do budowy dużych obszarów przewodzących (np. przy projektowaniu masy).
Istnieje możliwość zmiany odległości między poszczególnymi elementami elektronicznymi. Zmiana ta jest dokonywana wzdłuż osi pionowej lub poziomej, przy zachowaniu wprowadzonych wcześniej połączeń. Podczas projektowania na ekranie monitora ukazuje się obraz 2"x4", (50,8x101,6 mm), będący częścią powierzchni roboczej 10"x16". Użycie kursora pozwala na przesuwanie obrazu i oglądanie kolejnych części projektu. Możliwe jest także wyświetlanie całej powierzchni roboczej w zmniejszeniu.
Rysunek każdej warstwy zaopatrzony jest w następujące dane:
- skala rysunku
- data i czas
- tytuł projektu
- wersja programu SMARTWORK
- liczba otworów w płycie
- określenie warstwy (górna, dolna)
- wymiary płyty.

Program SMARTWORK wprowadzany jest do komputera z napędu dysków "A":. Dyskietka musi pozostawać w napędzie przez cały czas sesji roboczej. Po zgłoszeniu się programu należy załadować opcję EDIT.EXE, umożliwiającą utworzenie nowego pliku roboczego lub załadowanie już istniejącego. Plik roboczy, stanowiący projekt obwodu, może znajdować się na dyskietce lub dysku sztywnym. Konwersacyjna praca projektanta z komputerem jest bardzo przejrzysta, gdyż komunikaty ukazujące się na ekranie informują na bieżąco o stanie programu, możliwościach realizacji zleconych zadań, ewentualnie błędach popełnianych przez operatora.

Barbara Andrzejewska-Płatek

komputera

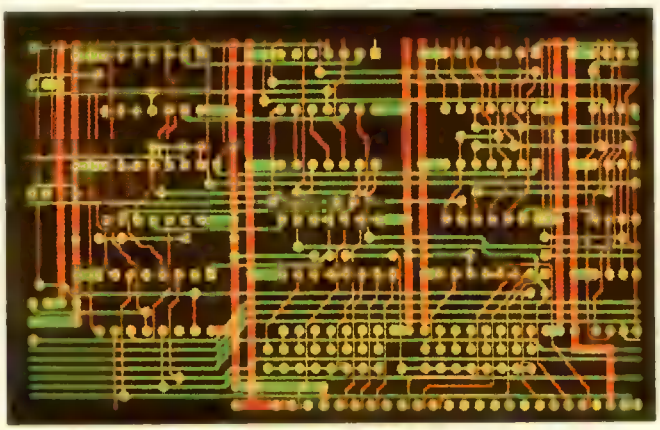
płytek o dużej „gęstości" upakowania automatycznie zostanie zrealizowanych nie więcej niż 50% połączeń - reszta musi być wtedy wprowadzana przez operatora. Rozmieszczanie elementów rozpoczyna się od tych, które muszą znaleźć się w określonym miejscu na płytce (np. łączówki). Położenie pozostałych ustala się według „gęstości" połączeń między poszczególnymi elementami. Na ekranie można wyświetlić połączenia dla całej płytki lub tylko połączenia wybranego elementu z pozostałymi.
Po rozmieszczeniu elementów na płytce wprowadzane są parametry technologiczne - średnice otworów, szerokości ścieżek, itp. Tak przygotowana baza danych pozwala na rozpoczęcie automatycznego prowadzenia ścieżek. Jednak w praktyce ścieżki zasilania i masy często wprowadzane są na wstępie przez operatora, ponieważ ich rozmieszczenie nie jest zbyt dobrze wykonywane przez komputer i wymaga korekty.
W programie automatycznego prowadzenia ścieżek można zdefiniować parametry:
- maksymalny czas realizacji pojedynczej ścieżki,
- maksymalną liczbę przelotek dla jednego połączenia,
- maksymalne odejście ścieżek w pionie i poziomie od optymalnego ułożenia,
- grubość ścieżek,
- parametr upakowania (mały, średni, duży),
- liczba pętli powtórnej realizacji dla ścieżek wcześniej odrzuconych,
- rodzaj płyty (cyfrowa lub analogowa).

Dla płyt z elementami cyfrowymi program żąda określenia numeru warstwy, na której mają być umieszczane ścieżki poziome. Ma to zasadniczy wpływ na liczbę mostków powstających przy lutowaniu na fali. System CAD 80/10 przewiduje możliwość wykorzystania do 64 warstw druku.
Program automatycznego prowadzenia ścieżek daje zadowalające rezultaty dla płytek wielowarstwowych lub płyt o niewielkiej gęstości upakowania (do 14 układów DIL16 na dm^2). W praktyce wykonuje ok. 80% połączeń, a pozostałe prowadzone są przez operatora.
Projekt jest testowany przez programy kontrolne, które sprawdzają jego zgodność z wcześniej wprowadzonymi danymi (listą elementów i połączeń). Na tym etapie realizacji wykonywane są rysunki rozmieszczenia elementów z opisami (lub bez), rysunki ścieżek zasilających i sygnałowych, otworów itd. Taka przejściowa dokumentacja pozwala na skorygowanie ewentualnych błędów projektanta i operatora.
Ostatnią fazą pracy jest zrobienie dokumentacji technologicznej. Składa się ona zazwyczaj z klisz wykonywanych na fotoploterze i taśmy perforowanej, służącej do sterowania wiertarką numeryczną. Wprowadzając odpowiednie komendy można dobrać rodzaj dokumentacji do istniejących standardów technologicznych (ISO, EIA), a także wprowadzić odpowiednie znaczniki bazowe służące do pozycjonowania otworów i klisz w procesie produkcji. Dokumentacja technologiczna może być następnie zapisana na dyskietkach (w celach archiwalnych).
Czas realizacji opisanego cyklu można skrócić o ok. 25% poprzez wprowadzenie przygotowanej wcześniej listy elementów i połączeń. W tym przypadku nie można jednak uzyskać rysunku schematu ideowego, większe jest też prawdopodobieństwo błędów. Dalsze skrócenie czasu pracy systemu (kolejne 25%) możliwe jest przez zastosowanie metody digitalizacji. Wymaga ona jednak wykonania całego projektu przez konstruktora - komputer przejmuje tylko prace dokumentacyjne.* Metoda ta jest stosowana przy projektach o wielu szczegółach mechanicznych lub dużym stopniu upakowania elementów.
System CAD 80/10 powstał około 5 lat temu i jest stale unowocześniany. W kraju pracuje obecnie siedem instalacji CAD 80/10, dwie lub trzy - systemu Quest oraz kilka systemów PCAD, EE Designer i Redac - Recall. Z rozwiązań nieprofesjonalnych należy wymienić Platine na Commodore 64 i różne wersje programu Smartwork dla IBM PC.

*) Rozważania te mają sens w polskich warunkach, tzn. gdy czas pracy systemu jest znacznie droższy od czasu pracy konstruktora - tu pomijanego (przypis red.)

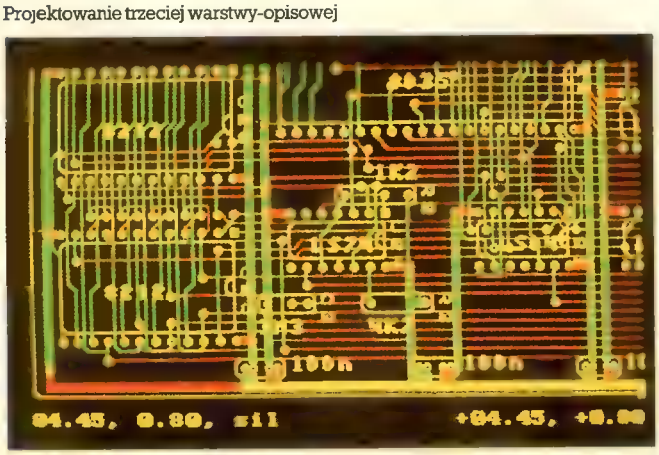
Adam Kutszyński

Platine 64



Fragment projektu płytki dwuwarstwowej

Projektowanie trzeciej warstwy-opisowej



Ze statystyk wynika, że Commodore C64 należy do najpopularniejszych komputerów domowych. Skłania to producentów oprogramowania do opracowywania rozwiązań, w których jego możliwości sprzętowe są wyjątkowo efektywnie wykorzystywane. W rezultacie, niektóre z oferowanych programów mają cechy, jakich oczekuje się od rozwiązań przeznaczonych do celów profesjonalnych. Jako przykład można tu podać program Platine 64. Służy on do projektowania płytek drukowanych o rozmiarach do 160x100 mm (przy rastrze 1,27 mm). Na płytce można zrealizować ponad 1500 połączeń. Użytkownik ma do dyspozycji ok. 140 funkcji, umożliwiających m. in. tworzenie biblioteki podzespołów, wykazów elementów i połączeń, rozmieszczanie elementów na płytce, automatyczne i „ręczne" prowadzenie ścieżek, zapis danych na dysku i wydruk projektu płytki.
Pracę rozpoczyna się od utworzenia biblioteki podzespołów - definiuje się wszystkie rodzaje używanych elementów i ułożenie ich wyprowadzeń. Następnie ustalany jest wykaz elementów, wprowadzanych z biblioteki i punktów lutowniczych nie należących do żadnego układu (np. doprowadzenia zasilania, sygnału itp.).
Poszczególne elementy i punkty lutownicze należy rozmieścić posługując się joystickiem, myszą lub klawiszami kursora. W kolejnym etapie projektowania podaje się informację o połączeniach między elementami i punktami lutowniczymi. Program ma specjalną funkcję „połączenia szynowe", która automatycznie wpisuje do wykazu połączenia wszystkich wyprowadzeń o tej samej nazwie - jest to użyteczne przy projektowaniu płytek systemów mikrokomputerowych. Najważniejszą funkcją programu jest projektowanie połączeń (ścieżek), które wykonuje procedura Autorouter (nawet przy skomplikowanych układach poprowadzonych zostaje ponad 90% ścieżek). Projektuje on również połączenia przewodem, kontakty dwustronne i połączenia pośrednie. W każdej chwili użytkownik może przerwać automatyczne prowadzenie ścieżek i sam poprowadzić ścieżkę.
Projekt płytki (dowolną jej stronę, maskę lutowniczą, a także współrzędne otworów dla wiertarki sterowanej numerycznie) można zobrazować przy użyciu drukarki lub plotera, przy czym - dzięki możliwości definiowania parametrów - program może współpracować nawet z niestandardowymi urządzeniami.
Platine 64 wypełnia wiele funkcji wymaganych od systemu projektowania obwodów drukowanych. Jednak możliwości programu (zwłaszcza maksymalne rozmiary płytki i brak wyjścia na fotoploter czy perforator taśmy) nie są wystarczające do zastosowań profesjonalnych.

Arkadiusz Jasik

Alternatywa dla sieci

W amerykańskich przedsiębiorstwach rozpowszechnia się nowa forma łączenia komputerów osobistych - zamiast organizować sieci LAN stosuje się tzw. automatyczne przełączniki danych.
Przełącznik danych jest wolno stojącym urządzeniem spełniającym podobną funkcję, jak punkt centralny w sieci o topografii gwiazdy. Pierwszym i najpopularniejszym zastosowaniem tych urządzeń było zapewnienie kilku użytkownikom dostępu do jednej drukarki. Komputery i drukarka były dołączone do ADS, który „kierował ruchem", buforując odbierane dane i zarządzając kolejką zgłoszeń. Nowe możliwości zastosowań powstały z chwilą wyposażenia ADS w funkcje umożliwiające przesyłanie i współdzielenie plików oraz wspólne użytkowanie dowolnych urządzeń peryferyjnych. Przykładem takiego urządzenia może być Net-Commander firmy Digital Products, pozwalający na jednoczesną komunikację 30 komputerów i urządzeń peryferyjnych.
Komputery połączone ze sobą przy pomocy ADS nie wymagają żadnych kart rozszerzających - dołączane są do przełącznika przez standardowe interfejsy szeregowe RS 232C. Przełącznik nieustannie nadzoruje wszystkie linie. ■ po wykryciu znacznika początku transmisji kieruje dane zgodnie z informacjami zawartymi w nagłówku przesyłanego bloku. Szybkość transmisji może wynosić od 300 do 19 200 bitów na sekundę.
Możliwe jest także przyłączenie ADS do dowolnej sieci lokalnej dysponującej portem sprzęgającym dla RS 232C (tzw. NIU - Network Interface Unit). Mogą to być np. sieci IBM Token Ring lub AT&T StarLAN. Taka konfiguracja jest często nazywana podsiecią (ang. subLAN).
Systemy oparte na automatycznych przełącznikach danych nie nadają się do zastosowań wymagających wielodostępnych baz danych, przesyłania dużych plików czy wspólnego użytkowania dysków sztywnych - do tych celów niezbędne są znacznie większe szybkości transmisji. Istotne są też ograniczenia odległości między łączonymi w ten sposób urządzeniami.
Głównym zastosowaniem ADS powinny więc stać się instalacje biurowe, pozwalające na komunikację i wspólne użytkowanie urządzeń peryferyjnych. Urządzenia te nie stanowią więc zagrożenia dla sieci LAN, pozwalają natomiast - same lub wspólne z LAN-ami - tworzyć konfiguracje połączeniowe w sposób elastyczny i lepiej niż dotychczas dostosować się funkcjonalnie do potrzeb użytkowników. **PC Week (JaW)**

Uniwersalny środek telekomunikacji

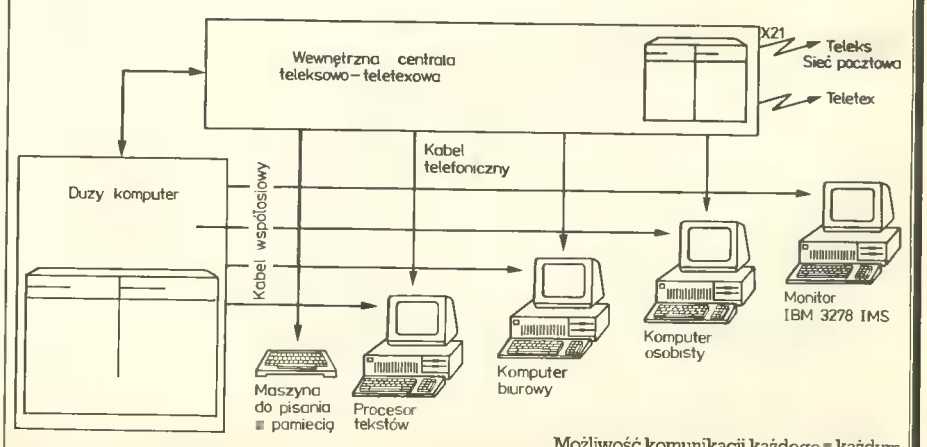
Teletex

Teletex początkowo miał spełniać rolę szybkiego dalekopisu z rozszerzonym zestawem znaków. W ostatnich latach zakres jego zastosowań znacznie się jednak zwiększył. Z teletexu można korzystać na skomputeryzowanych stanowiskach pracy biurowej, uzyskując w ten sposób możliwość przetwarzania przyjmowanych danych.
Teletex, w odróżnieniu od teleksu, nie wymaga stosowania specjalnych urządzeń końcowych (np. dalekopisu). Każde stanowisko pracy wyposażone w maszynę do pisania z pamięcią, komputer osobisty lub terminal można dzięki teletexowi przekształcić w wielofunkcyjne urządzenie końcowe sieci telekomunikacyjnej.
Wykorzystywany do pracy biurowej komputer osobisty, jeśli nie zostanie włączony w system komunikacyjny, jest tylko nieco lepszą maszyną do pisania. Konieczność wielokrotnego wprowadzania informacji (ze względu na brak kompatybilności urządzeń, które przetwarzają ją w różny sposób) bardzo ogranicza korzyści wynikające ze zwiększonej szybkości przetwarzania.
Każde przedsiębiorstwo ma określoną infrastrukturę i doświadczenia w komunikacji telefonicznej za pośrednictwem wewnętrznych central oraz w komunikacji dalekopisowej. Istniejącą już bazę warto dalej rozwijać przy tworzeniu systemów komunikacji wewnątrzzakładowej. Doskonale nadaje się do tego właśnie teletex, gdyż łączy automatyzację pracy biurowej z telekomunikacją. Umożliwia nie tylko komunikację zewnętrzną, ale także - w połączeniu z cyfrową centralą zakładową - komunikację wewnętrzną. Można przy tym przesyłać informacje w różnych postaciach, obok tekstów również dane i obrazy graficzne. Przy przesyłaniu tekstów teletex oferuje większe możliwości niż inne metody komunikacji, gdyż za jego pośrednictwem może wymieniać dane pokaźna liczba programów przetwarzających teksty. Ponieważ użytkownicy takich programów domagają się możliwości korzystania z teletexu - producenci modyfikują warstwę komunikacyjną. Są już dostępne adaptacje m. in. PCtext3, MS-WORD, czy TEXAS-WINDOWS, pozwalające na współpracę ze sterownikiem teletexu.
Teletex został też zintegrowany z innymi pakietami oprogramowania, umożliwiając m. in. drukowanie formularzy czy zdalną „konwersację" programów.
Oddzielenie funkcji transmisyjnych od przetwarzania umożliwia realizację systemu poczty elektronicznej. Integralną część teletexu stanowią funkcje: przesyłanie okólne, przesyłanie na żądanie, ochrona przed zapisem i przed odczytem oraz przesyłanie pakietów danych.
Kierunkiem rozwoju teletexu są instalacje sieciowe. Na początku wprowadzane są sterowniki pracujące w trybie wielodostępu. Umożliwiają one przyłączenie do jednego zewnętrznego łącza wielu lokalnych użytkowników, a nawet utworzenie prostej sieci wewnętrznej. Są już zrealizowane wejścia do dużych systemów komputerowych (np. IBM 36, 38, Siemens BS 2000).
Integracja teletexu z telefonią cyfrową jest następnym krokiem, przybliżającym utworzenie uniwersalnego systemu poczty elektronicznej.

Co z ISDN?

ISDN, czyli cyfrowa sieć zintegrowanych usług, polega na transmisji w postaci cyfrowej rozmów, obrazów i danych (mikroklan 7/87). ISDN zapewnia usługi na wyższym poziomie technicznym niż teletex, lecz wciąż jeszcze znajduje się w fazie próbnej. Można przypuszczać, że przewidywany czas użytkowania już istniejących sieci będzie o wiele dłuższy, niż to wcześniej planowano.
Opóźnianie wprowadzania ISDN byłoby jednak szkodliwe także dla teletexu, który znacznie zyska na przejściu na zintegrowaną łączność cyfrową. Teletex będzie mógł wtedy w pełni wykorzystać dużą szybkość transmisji w sieci ISDN. Możliwosci teletexu nie będą ograniczane, ponieważ jego protokół jest już ujęty w standardach ISDN.
Sterownik teletexu jest w RFN tanszy od dalekopisu i kosztuje ok. 4550 DM.
Oszczędność staje się jeszcze bardziej widoczna przy zainstalowaniu cyfrowej centrali wewnętrznej. Przy 40 urządzeniach końcowych teletexu nakłady spadają do 1000 DM na stanowisko robocze. Duża szybkość przesyłania sprawia, że dwa łącza teletexowe, wymagane przez pocztę RFN jako minimum przy takiej konfiguracji, zupełnie wystarczą do obsłużenia całego ruchu. Przyłączenie teletexu do sieci kosztuje 220 marek miesięcznie.
W dłuższym okresie decydującą rolę odgrywają jednak opłaty za połączenia. Koszt przesłania tekstu formatu A4 wynosi w RFN 3,5 marki za transmisję teleksową i tylko 0,20 marki za transmisję teletexową. Niższą opłatę (50 fenigów) można uzyskać przesyłając teleks po godzinie 18. W przypadku teletexu transmisja zostanie wykonana automatycznie w godzinach nocnych za 6 do 8 fenigów. W tej sytuacji opłacalne jest przesyłanie listów za pośrednictwem teletexu.
Dla zwiększenia walorów ekonomicznych teletexu przyjęto, że każdy użytkownik będzie miał dostęp do całej światowej sieci teleksowej. Tempo wzrostu liczby nowych abonentów teletexu w zasadzie przekroczyło już swój punkt kulminacyjny. Natomiast wciąż rozszerza się zakres oferowanych przezeń usług.

Opr. **Andrzej Suwała** wg micro 6/87



Możliwość komunikacji każdego ■ każdym

Tworzenie obrazu jest sztuką bez względu na to, czy odbywa się na płótnie malarskim czy na monitorze graficznym. Obowiązujące reguły i środki wyrazu są takie same: kolor, światło, cień, szarość, refleks - inne są jedynie narzędzia i techniki posługiwania się nimi. Część tajników warsztatowych grafiki komputerowej wiąże się z odwzorowaniem na płaskim ekranie trzeciego wymiaru.

Trzeci wymiar

Przekazywanie informacji w formie obrazu jest wykorzystywane przez człowieka od początku jego rozumnej egzystencji. Jest przy tym oczywiste, że informacja w tej postaci jest odbierana szybciej i pełniej niż w formie ciągu znaków pisarskich. Fakt ten powoduje niezwykle dynamiczny rozwój grafiki komputerowej. W technice komputerowej stosunkowo łatwo tworzy się obrazy obiektów dwuwymiarowych, takich jak figury płaskie, histogramy, wykresy funkcji jednej zmiennej itp. Natomiast duże kłopoty sprawia odwzorowanie trzeciego wymiaru. Algorytm grafiki trójwymiarowej (ang. 3D graphics) tzn. poziomicowania, rysunku perspektywicznego, przesłaniania, przenikania, operowania światłem i cieniem są w większości wypadków trudno dostępne lub utajnione. Stąd także literatura na ten temat jest uboga i zawiera mało konkretnych informacji. Prezentowany artykuł poświęcony jest pierwszemu z wymienionych zagadnień, tzn. technikom poziomicowania.

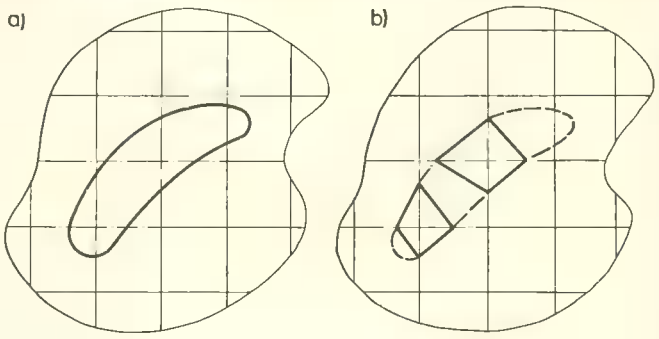
Algorytmy poziomicujące

Jedno z najczęstszych zadań, w którym spotykamy się z koniecznością przedstawienia trzeciego wymiaru na płaszczyźnie, polega na narysowaniu wykresu funkcji dwóch zmiennych $f=f(x,y)$. Gdy zależy nam na otrzymaniu dokładnych informacji o jej zmienności, nieodzowne jest uzyskanie obrazu poziomic funkcji, czyli linii stałej wartości. Wykres taki nazywamy mapą funkcji. Najprostsza metoda kreślenia mapy polega na podzieleniu badanego obszaru zmienności funkcji na m wierszy i n kolumn, czyli m*n prostokątów, wyliczeniu wartości funkcji w węzłach powstałej w ten sposób siatki i analizie kolejnych pól. Analiza taka sprowadza się do poszukiwania w prostokącie dwóch boków przecinanych poziomicą o wartości f_0, innymi słowy takich, dla których jest spełniony warunek $(f(A)-f_0)*(f(B)-f_0)\leq 0$ (punkty A i B są końcami analizowanego boku prostokąta). Położenie punktu przecięcia z poziomicą f_0 najłatwiej jest określić poprzez liniową aproksymację funkcji na odcinku AB. W przypadku znalezienia w analizowanym prostokącie co najmniej dwóch takich punktów, łączy się je z odcinkiem. Procedurę tę należy powtórzyć dla kolejnych wartości poziomic w każdym z elementarnych pól.

Metoda ta daje zadowalające rezultaty dla funkcji łagodnie zmiennych o nieskomplikowanym układzie poziomic, gdy niezbędna gęstość siatki nie przekracza 30 na 30 elementów. Jednak w przypadku, gdy warunek ten nie jest spełniony, wraz z gęstością siatki wzrasta liczba punktów obliczeniowych i liczba kreślonych wektorów, a tym samym wydłuża się nieproporcjonalnie czas rysowania mapy. Szczególnie niekorzystny jest on w przypadku, gdy analizowana funkcja przedstawiona jest w postaci sumy szeregu, gdy rysunek powstaje na ploterze. Dodatkowo w powyższym algorytmie na uzyskany efekt nakładają się błędy aproksymacji przedstawione na rys.1.

Chcąc skonstruować algorytm pozbawiony tych wad, odejdźmy na moment od matematyki. Wybierzmy w naszym otoczeniu dowolny przedmiot. Niech będzie to na przykład aparat telefoniczny. Prowadząc uważnie wzrok wzdłuż jego obrysu spostrzeżemy, że nasze oko wykonuje ruch odcinkami, a ich długość zależy od krzywizny badanej linii. W przypadku, gdy zmienia się ona w sposób gwałtowny, na przykład: na krawędzi, zatrzymujemy wzrok w celu znalezienia nowego kierunku. Zdarzyć się może, że gdy będziemy zbyt szybko przemieszczać nasze spojrzenie, zgubimy na moment badany kontur. Wtedy cofamy wzrok i powtórnie wolniej analizujemy trudniejszy odcinek. W procesie tym oko dostosowuje szybkość analizy obrazu zarówno do krzywizny linii składających się na niego, jak i do charakteru ich zmienności. Jak łatwo się domyślić, w oparciu o te same zasady działać powinien efektywny algorytm poziomicujący. Wyróżnić w nim można dwa zasadnicze etapy. Celem pierwszego, który nazywamy przeszukiwaniem wstępnym, jest zlokalizowanie poziomicy, a dokładniej określenie położenia jej dowolnego punktu oraz lokalnego kierunku. Etap drugi polega na śledzeniu poziomicy, tzn. na znajdowaniu współrzędnych kolejnych jej punktów. Odległość pomiędzy nimi nazywana krokiem metody, powinna zapewnić wystarczającą gładkość otrzymanej łamanej. Pamiętać bowiem należy, że w każdym kroku dwa kolejne punkty są łączone na rysunku odcinkiem linii prostej. Z tego względu metodę tę nazywamy metodą śledzącą pierwszego stopnia.

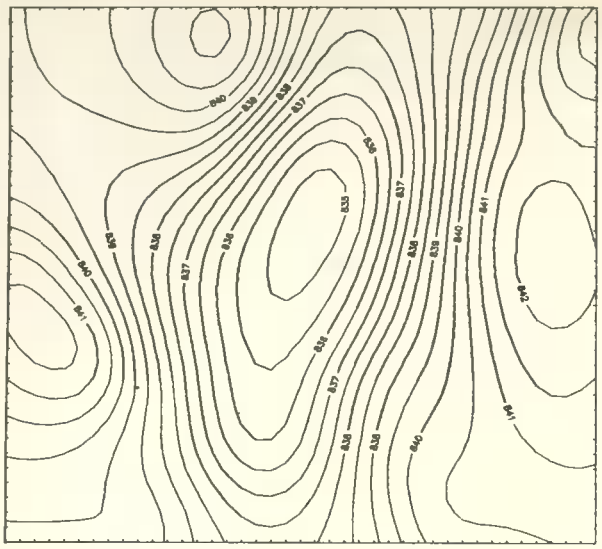
Poszukiwanie punktu startowego zaczynamy od podzielenia analizowanego obszaru zmienności funkcji na m*n prostokątów. Przeszukiwanie wstępne polega wtedy na sprawdzeniu, czy kolejne pola są przecięte daną poziomicą. Czynimy to analogicznie, jak w opisanym wcześniej algorytmie z tą jednak różnicą, że zbadane pola zaznaczamy. Jak się bowiem często zdarza, w badanym obszarze występuje nie jedna, lecz kilka poziomic o zadanej wartości f_0. Odróżnić je można zaznaczając pola w trakcie przeszukiwania wstępnego oraz śledzenia (rys. 3). Gdy zostanie znaleziony bok



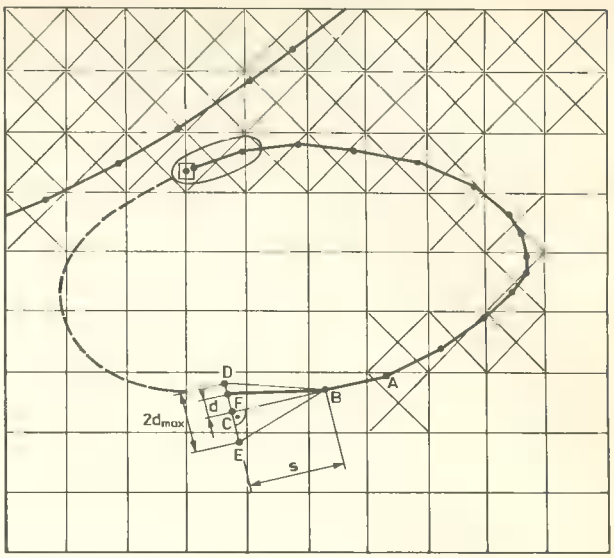
1. Błędy aproksymacji:a)rzeczywisty przebieg poziomicy; b)efekt działania aproksymacji

przecinany poszukiwaną poziomicą, należy dokładnie określić punkt przecięcia (tzn. z błędem mniejszym niż 0,001 długości boku elementarnego prostokąta). Nie wystarczy w tym celu zastosowanie aproksymacji liniowej. Konieczne jest rozwiązanie równania $f(x,y)=f_0$ przy ustalonej wartości x lub y, odpowiedniej dla danego boku. Możemy tu zastosować na przykład metodę bisekcji, siecznych lub inną. W celu określenia lokalnego kierunku poziomicy, otaczamy punkt startowy kwadratem o długości boku wielokrotnie mniejszej od długości boku elementarnego prostokąta. Powinna ona być jednak większa od błędu, z jakim określone zostały współrzędne punktu startowego. Położenie drugiego punktu poziomicy obliczyć możemy wykorzystując aproksymację liniową dla jednego z przecinanych boków kwadratu.

Technika śledzenia poziomicy opiera się na założeniu, że kolejny jej punkt należy do linii prostej zdefiniowanej przez dwa poprzednie. Założenie to jest prawdziwe w przypadku, gdy sama poziomica jest linią prostą. Najczęściej jednak tak nie jest, a kolejny punkt, oznaczony na rys. 3, literą F, leży w odległości d od analizowanej prostej. W celu wyznaczenia jego położenia należy na odcinku DE zastosować aproksymację liniową lub lepiej kwadratową. W drugim przypadku spośród dwóch rozwiązań wybrać należy to, które



2. Mapa funkcji



3. Technika sledzenia

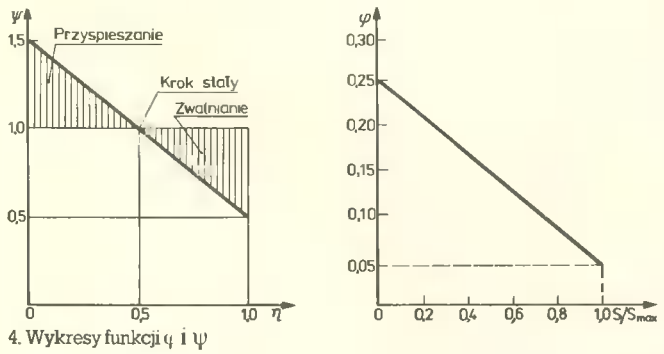
jest bliższe punktowi C. Za miarę odchyłki d możemy przyjąć parametr dewiacji η określany wzorem $\eta = d/d_{max}$, gdzie d_{max} jest związana z aktualnym krokiem s zależnością $d_{max}\varphi s$. W przypadku poziomic łagodnie zmiennych, tzn. gdy $\eta \in \langle 0,1\rangle$, długość kolejnego kroku s_{n+1} wynosi $s_{n+1}=\psi s_n$, gdzie parametr ψ jest zależny od dewiacji. Zdefiniowane w ten sposób funkcje φ oraz ψ mają decydujący wpływ na gładkość otrzymanej łamanej oraz efektywność programu. Ich interpretacja jest prosta - funkcja ψ odpowiada za zwiększenie lub zmniejszenie kroku w zależności od wartości krzywizny, natomiast funkcja φ decyduje o czułości na zmiany krzywizny. Kształty funkcji φ i ψ zaproponowane na rys. 4 nie są obligatoryjne. Zostały one dobrane eksperymentalnie i spełniają dobrze swoje zadanie przy śledzeniu elips o maksymalnym stosunku półosi 20:1. W praktyce jest to wartość zupełnie wystarczająca. W przypadku, gdy parametr dewiacji η jest większy od jedności, należy zmniejszyć krok s na przykład dwukrotnie. Jeżeli mimo tego nawet dla $s=s_{min}$ warunek $\eta \in \langle 0,1\rangle$ nie jest spełniony, w celu zapewnienia dostatecznej gładkości poziomicy, należy cofnąć się do kroku poprzedniego, zmniejszyć go i powtórzyć obliczenia.

Wyjaśnienia wymaga kwestia dobrania zakresu początkowej wartości kroku oraz warunku zaprzestania śledzenia. Ogólnie wartość kroku powinna zawierać się w granicach 0,01 do 1,0 wymiaru oczka siatki. Stosowanie mniejszych wartości nie poprawia gładkości poziomicy i niepotrzebnie zwalnia wykonywanie programu, natomiast większe wartości komplikują poważnie technikę zaznaczania elementarnych prostokątów, przez które przechodzi analizowana poziomica. Szukając wartości początkowej kroku, należy zacząć od wartości najmniejszej, a następnie zwiększać ją aż do momentu, gdy parametr dewiacji η zbliży się do 0,5, co odpowiada ustalonej długości kroku. W przypadku, gdy w trakcie śledzenia osiągnięta zostanie granica badanego obszaru, należy powrócić do punktu startowego i kontynuować śledzenie w kierunku przeciwnym. Równie często zdarza się także, że poziomica jest krzywą zamkniętą. Jej zamknięcie powinno nastąpić po dotarciu do punktu startowego. Nie jest to jednak zadanie łatwe. Proste sprawdzanie odległości i zatrzymanie się, gdy jest ona mniejsza od zadanego ε, prowadzi do przypadkowego zamykania linii lub krążenia w koło. Dobre efekty daje sprawdzanie sumy odległości od drugiego i trzeciego punktu, które są w takim wypadku ogniskami elipsy. Jej dłuższa półoś jest równoległa do lokalnego kierunku poziomicy. Wynika stąd, że proces powinien zostać zatrzymany, gdy kolejny z punktów znajdzie się wewnątrz zdefiniowanej w ten sposób elipsy.

Jeśli w wyniku przeszukiwania wstępnego oraz śledzenia pozostaną jeszcze nie zbadane elementarne prostokąty, należy kontynuować przeszukiwanie w celu znalezienia następnych poziomic. Po przeanalizowaniu wszystkich pól, algorytm ten należy powtórzyć dla kolejnych wartości poziomic.

Przedstawiona procedura oparta jest na metodzie pierwszego rzędu, tzn. liniowej ekstrapolacji kształtu poziomicy i schematu dwupunktowego. Oznacza to, że do znalezienia kolejnego punktu jest potrzebna znajomość dwóch poprzednich. W pewnych zastosowaniach, szczególnie wtedy gdy obliczenie wartości funkcji jest wysoce czasochłonne, konieczne okazuje się użycie schematu trzypunktowego. W schemacie tym kolejnego punktu poszukuje się na przedłużeniu łuku koła opartego na trzech poprzednich. Kolejnych punktów nie łączy się wówczas jednym odcinkiem, lecz łamaną. Postępowanie takie pozwala na wydłużenie kroku i przyspiesza działanie programu (jest on jednak wtedy bardziej skomplikowany). Problem czasochłonnych funkcji można prościej ominąć, stosując dla każdego elementarnego prostokąta aproksymację funkcji wielomianem, który w najprostszej postaci można zapisać $f^*(x,y)=Axy+Bx+Cy+D$. Stałe A, B, C, D oblicza się z warunku $f=f^*$ dla punktów narożnych. Wielomian ten ma duże zastosowanie, mimo że nie zapewnia ciągłości pochodnej funkcji na granicy pomiędzy prostokątami. Wykorzystuje się go także, gdy funkcja f nie jest dana w postaci analitycznej, a jedynie jako zbiór wartości w węzłach siatki.

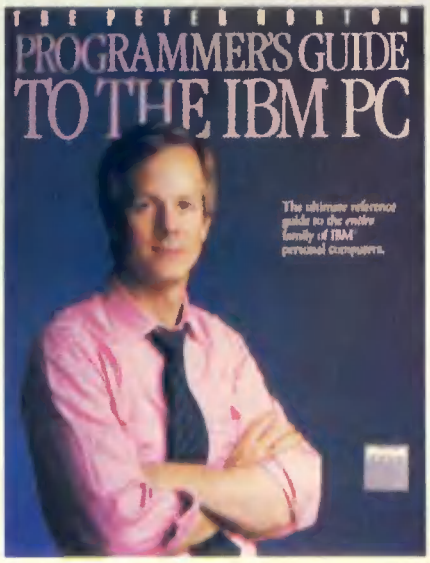
Pełna postać przedstawionej procedury stosowana jest w pakietach ogólnego przeznaczenia. W wielu jednak zastosowaniach dysponujemy często poważną wiedzą o charakterze zmienności funkcji. Stosować wtedy możemy schematy prostsze, np. o stałej długości kroku, bez przeszukiwania itp. Program taki można stosunkowo łatwo napisać samemu na mikrokomputerze klasy Commodore 64 lub nawet prostszym, ponieważ algorytm w tej postaci nie zajmuje dużo pamięci. Długość kompletnego programu napisanego zgodnie z przedstawioną procedurą wynosi 500...600 linii. Z tego powodu procedura poziomicująca, zastosowana w handlowym pakiecie PLOTCALL dla komputera typu IBM PC, oparta jest na pierwszym - najprostszym z zaprezentowanych algorytmów. Za jej pomocą został uzyskany wykres przedstawiony na rys. 2. Stosunkowo dobra gładkość linii wynika z dużej gęstości siatki. Wynosiła ona 40 na 40 elementów. Wpłynęło to jednak poważnie na efektywność obliczeń. Pakiet ten, umożliwia opisywanie poziomic i współpracuje z szeregiem najpopularniejszych drukarek. Przedstawiony wykres uzyskany został na drukarce STAR SG-10. Jego wysoką jakość osiągnięto dzięki trzykrotnemu przebiegowi głowicy w każdym wierszu.



4. Wykresy funkcji φ i ψ

Robert Moldach

Elementarz dla profesjonalistów



Zadziwiająca to książka - ciekawa zarówno dla doświadczonych, jak i początkujących programistów używających komputerów osobistych IBM. Wiadomości są podane w sposób systematyczny i przystępny. Doświadczony programista znajdzie tu jednak obszerny zestaw informacji zwykle rozproszonych w wielu książkach. A więc, koniec z wertowaniem stosu literatury!

W książce zebrano dane dotyczące wszystkich komputerów osobistych rodziny IBM: PC, XT, AT, PCjr i Portable - co daje możliwość szybkiej oceny przydatności danego typu komputera do zadanych funkcji. Znacznie czytelniej niż w firmowej literaturze IBM zostało wykonane zestawienie informacji o odwołaniach do portów, procedur BIOS-a i funkcji DOS-a.

Mimo nagromadzenia informacji książka nie jest encyklopedią ani słownikiem terminów komputerowych, przeciwnie - zawiera materiał znacznie poszerzony w stosunku do innych publikacji, a czasami wiadomości nigdzie nie publikowane. Częściowo dotyczy to ciekawostek. Często są to spostrzeżenia doświadczonego programisty sygnalizujące błędy (i pomagające je ominąć). Książka jest napisana zajmująco, tak żywym językiem, że można ją czytać niemal jak powieść sensacyjną.

Najsłabszy jej punkt stanowi zakończenie. Ostatnie rozdziały - omawiające języki programowania i budowę programów - napisane ogólnikowo, a znajdujące się tam informacje są oczywiste dla tych, którzy rozumieli poprzednie rozdziały. Dodatek poświęcony instalowanym programom sterującym (Installable Device Drivers) jest nieporozumieniem - nie wnosi nic konkretnego. Do wad należy również przemilczenie popularnych rozwiązań oferowanych przez firmy nie związane z IBM (np. karta Hercules). Informacje dotyczące komputera AT nie zawsze są ścisłe (np. w zestawieniu procedur obsługi przerwań BIOS - opis przerwania 15H). Trzeba jednak uwzględnić fakt, że książka została napisana w 1985 r. Wspomniane wady są rekompensowane przez znakomicie opracowane partie książki omawiające obsługę systemu dysków, klawiatury i sterowników obrazu. Każdemu z tych tematów poświęcone są po dwa rozdziały dotyczące podstaw programowania na najniższym poziomie i dedykowanym procedurom BIOS-a. Silną stroną książki jest też dogłębny opis funkcji systemu operacyjnego DOS wersje 1.00...3.00

Ogólnie można powiedzieć, że książka przedstawia sprzęt widziany oczyma programisty, choć nie wynika to bezpośrednio z układu materiału. Krótki opis układów scalonych, z których zbudowany został komputer, nie powinien zrazić nawet informatyków nieufnie traktujących sprawy sprzętu. Lektura książki może nawet stanowić zachętę do poznania tajników budowy wnętrza komputera.

Każdy z rozdziałów (a czasem podrozdziałów) stanowi zamkniętą całość. Po przejrzeniu książka nie trafi na półkę - warto do niej sięgnąć ilekroć istnieje potrzeba niekonwencjonalnego wykorzystania możliwości monitora, napędów dysków, generatora dźwięków itp. Przy programowaniu w języku maszynowym jest bardziej przydatna niż „DOS Technical Reference" i „PC (XT/AT) Technical Reference".

Sprowadzane z Tajwanu komputery najczęściej przysyłane są bez lub z bardzo skąpą dokumentacją. Niektóre z firm, na specjalne życzenie zamawiającego, dostarczają odbitki dokumentacji IBM w cenie zbliżonej do oryginału omawianej książki. W tej sytuacji warto, by książka Petera Nortona zyskała sobie status „wyposażenia" dla tych komputerów, które przeznaczone są do poważniejszych prac programistycznych.

Maciej A. Markowski

Norton Peter. The Peter Norton Programmer's Guide to the IBM PC,
Bellevue, WA: Microsoft Press, 1985, $19,95, 426 stron.

Uwaga: Książki wydawnictwa Microsoft Press są w Europie dystrybuowane przez Penguin Books Ltd. Mimo że w zasadzie nie jest prowadzona detaliczna sprzedaż wysyłkowa, uzyskaliśmy obietnicę p. Davida Williamsa (dyrektora oddziału europejskiego), że dla czytelników **mikro**klanu zostanie uczynione odstępstwo. Szczegółowe informacje w tej sprawie można uzyskać pisząc (w j. angielskim) pod adres: Penguin Books Ltd., Bath Road, Harmondsworth, Middlesex UB7 ODA, Great Britain.

W najbliższych numerach **mikro**klanu zamieścimy recenzje kolejnych książek nadesłanych nam przez Penguin Books: „Running MS-DOS" i „Supercharging MS-DOS".

(red)

daton oferuje daton poleca daton

Sp. z o. o.
Biuro Techniczno-Handlowe
ul. Waliców 19/20
00–865 Warszawa
tel. (022) 242659
tlx. 812729 DATON PL

● systemy IBM XT/AT w dowolnej konfiguracji – w cenie sprzętu oprogramowanie narzędziowe i szkolenie użytkowników
● pakiet funkcjonalny do szybkich obliczeń numerycznych hardware: CPU MC68020 (32 bity), FPU MC68881, RAM 1MB oprogramowanie: FORTRAN, C, PASCAL, BASIC
● urządzenia peryferyjne:
– monitory (monochromatyczne, kolorowe, EGA)
– digitizery
– plotery
– streamery
– twarde dyski (20, 30, 40, MB)
– drukarki "STAR"
– programatory EPROM
– floppy dyski, i inne
● sieci lokalne:
– trans-net
– advanced trans-net
– d-link
● instalacje i wdrożenia użytkowego oprogramowania:
– płace
– zus
– kadry
– fundusz mieszkaniowy
– kasa zapomogowo-pożyczkowa

Natychmiast wysyłamy oferty cenowe.
Na dostarczone systemy udzielamy rocznej gwarancji.
Wykonujemy również płatny serwis pogwarancyjny.
Termin dostawy sprzętu max. 1 miesiąc.

Spis treści 1986-1987

Budujemy sami

Interfejs szeregowy do ZX Spectrum. T.Cierpisz. 3/87 16.
Jak dołączyć 2048 portów do ZX Spectrum. K.Kontek. 1/86 24.
Katarynka. J.Orkiszewski, 3/86 23.
Klawiatura hallotronowa. J.Orkiszewski, I.Nowosielski. 3/86 15.
Konwerter Centronics RS-232C. S.Błeszyński. 1/87 20.
Programator pamięci EPROM. S.Gardyniuk. 1/86 18.
Rozszerzenie pamięci mikrokomputera ZX Spectrum. K.Amborski, D.A.Przygoda. 2/87 24.
Układ interfejsu do Atari 600XL i 800XL. M.Szczęch, K.Skonieczny. 1/87 III.

Eksploatacja

Bałagan na dysku.8/87 11.
Jak dbać o dyskietki?. 1/86 8.
Jakość dyskietek. 1/87 12, 2/87 30.
Przeglądanie plików. M.Karolczak. 8/87 28.
Przenoszenie plików. Z.Pojmański. 8/87 22.
Ratujmy nasze zbiory. M.A.Markowski. 9/87 21.

Języki programowania

BASIC dla początkujących. 1/86 2, 2/86 2, 3/86 2, 1/87 24, 2/87 27, 3/87 26, 4/87 30.
Szybciej, krócej, taniej... A.J.Piotrowski.9/87 2.
To C or not to C? A.J.Piotrowski. 9/87 2.
Turbo Basic. A.Nafalski, M.Wójtowicz. 8/87 10.
System Turbo C. J.Bielecki.9/87 4.
Uczmy się Pascala! 1/87 2, 2/87 7, 3/87 29, 4/87 27.

Komputery i ludzie

Bez stresów. 1/86 10.
Czytelnicy i komputery. 9/87 14.
Mniejsze promieniowanie. 6/87 23.
Nie ma jutra bez transputera.9/87 14.
Oczami nauczycieli. K Kontek. 2/86 25.
Poufność informacji. 3/87 23
Wojna przeciw piratom. 8/87 22.
Z komputerowego światka. 7/87 3.
Zabezpieczenie programów GEM. 1/86 3.
Zachód łagodzi kontrolę. 9/87 25.
Zaślepienie czy zacofanie? K.Isaak. 1/86 26.

Prezentacje sprzętu

32-bitowy AT. 5/87 10.
7 myszy. 5/87 24.
Akwizacja danych w tle. 6/87 31.
Alternatywa dla sieci. 9/87 28.
Amiga 500. 5/87 4.
Amstrad CPC 6126. 3/86 21.
Amstrad Joyce czyli przetwarzanie tekstów dla początkujących. 3/87 8.
Amstrad PC 1512. 3/87 12.
Amstrad story. M.Górecki. 1/86 14.
Apple II w sieci lokalnej. 2/86 31.
Atari 800 XL. J.Orkiszewski. 2/86 15.
CeBIT'87: nowe druki. 4/87 29.
Chyba jednak pseudostandard. J.Rudomina. 5/87 26.
Compaq Deskpro 386. 3/87 4.
Czy przewaga technologiczna wystarczy? 2/87 14.
Dodatkowa pamięć RAM do Amstrada. J.Młodzki. 3/87 22.
Drukarka laserowa o rewelacyjnej cenie. 4/87 13.
Drukarka strumieniowa. 1/86 7.
Drukarki NEC P6 i P7. 6/87 23.
Drukowanie barwne. 2/87 16.
Dysk stały dla każdego. 4/87 7.
Dyski typu Whintey. 2/86 31.
Hercules Plus. 5/87 26.
HP-Quiet-Jet-Plus. 4/87 11.
IBM. 5/87 7.
IBM PC/At. 1/87 7.
Imperium kontratakuje. Z.Pojmański. 7/87 7.
Jak zmienić PC w AT? 6/87 7.
Kameleon z dziewięcioma igłami. 4/87 2.
Komputer o dwóch obliczach. 5/87 2.
Komputer osobisty DIALOG DTC-8. K.Kamiński 3/86 12.
Komputer, jakiego nie było! G.Lander. 6/87 2.
Komputery IBM PC XT/AT. K.Kontek. 1/87 4.
Komputery osobiste rosną w siłę. 6/87 6.
MacIntosh II. A.J.Piotrowski 7/87 21.
Made in Germany. 6/87 29.
Minstrel 4. 6/87 31.
Monitor barwny Multisync. 4/87 15.
Nowa generacja mikrokomputerów IBM. 4/87 30.
Nowe komputery - BBC Master Series. 2/86 10.
Podwójny milioner. 8/87 6.
Powiew wielkiego świata. 6/87 19.
PS/2 nie oznacza końca klonów? 7/87 2.
Rewolucja w poligrafii. M.Jaworski. 8/87.
Sinclair 128 KB. 1/86 13.
Spokrewnione z wyboru czyli kompatybilne z własnej woli. 1/87 17.
Sprzętowy klucz. M.A.Markowski.9/87 21.
Sterowniki mikroprocesorowe. K.Kamiński. 6/87 13.
Stolik graficzny Cherry. 5/87 29.
System dyskowy MUEL-85. P.Parlewicz, A.Więckowski, G.Zawadzki. 3/86 26.
Systemy modularne w automatyce. S.Jagiełło, K.Rzymkowski. 5/87 8.
Szepcząca drukarka. 9/87 14.
Szybkość w walizce. 8/87 9.
Terminal 3D. 9/87 14.
Tylko 3 razy! 6/87 19.
Układ sprawdzający pisownię. 6/87 23.
Winchester. 1/86 3.
Wymazywalna pamięć optyczna. 6/87 19.
Z pozycji konstruktora. A.J.Piotrowski. 5/87 5. 6/87 4
Znów lord Sinclair. 5/87 29.

Programowe usprawnienia

AutoCAD drukowanie rysunków. R.Szuniewicz. 6/87 18.
Automatyczne wygaszanie obrazu. R.Rzędkowski. 5/87 30.
Drukowanie nietypowych znaków. 4/87 20.
Łaty na Wordstarze. R.Szuniewicz. 5/87 22.
Nakładanie łaty. 8/87 9.
Polskie znaki na IBM PC. R.Szuniewicz. 7/87 29.
Superlife. J.Szafranek. 2/86 11.
Tablica kontrolna systemu TV PAL. W.Świtalski. 2/87 6.
Turbotape 64. K.Dybowski, M.Silski. 2/86 13.
ZX Spectrum. A.J.Piotrowski. 1/86 30.

Programy użytkowe

1-2-3. R.Szuniewicz. 9/87 10.
AutoCAD. 1/87 9.
Bazy danych zmieniają oblicze. 8/87 6.
CHIWRITER program do redagowania tekstów technicznych. A.J.Piotrowski. 5/87 19.
dBase III plus, 8/87 2.
Komputerowe tłumaczenie. 6/87 6.
Komputerowy kumpel. M.A.Markowski. 6/87 6.
Łapacz myśli. 9/87 11.
Konstruowanie przestało być trudne. 3/86 5.
Młodszy brat SideKicka. 9/87 14.
Mniej znane możliwości asemblera GENS3. P.Parlewicz. 1/87 15.
Norton Commander. R.Rzędkowski. 6/87 22.
Nowości z firmy Borland. 6/87 30.
Olivetti 386 i ... 9/87 7.
Oprogramowanie Amigi. 5/87 4.
PEPSY - pakiet programowy do obliczeń statycznych. 2/87 4.
Platine 64. A.Jasik. 9/87 27.
Polscript. B.Seredyński. 4/87 18.
Puste biurko. 2/87 19.
Smartwork. B.Andrzejewska-Płatek. 9/87 26.
Spectrum kontra ... Macintosh. 1/86 7.
Wordstar 2000 czyli nowe szaty króla. 1/87 22.

Sprzęt od wewnątrz

CAMAC i inne. K.Rzymkowski. 9/87. 12
Emulator/ ... na przykład z firmy Microtek. A.J.Piotrowski. 8/87 31.
Generator znaków. J.Orkiszewski. 1/86 31.
Hercules. S.Piotrowski. 5/87 14.
Intel 80 386 4/87 4.
Magistrala MicroChannel. A.J.Piotrowski. 7/87 11.
Magistrala obiektowa PC XT/AT. K.Rzymkowski. 5/87 9.
Mikroprocesory MC 68000. P.Parlewicz. 2/87 2.
Mikroprocesory z serii MC 68000 (2). J.Orkiszewski, P.Parlewicz. 3/87 20.
Pamięć dynamiczna. P.Smólski. 3/86 20.
Pamięć zewnętrzna mikrokomputerów na mini-dyskach elastycznych (1). A.Pluta. 3/86 27.
Pamięć zewnętrzna na minidyskach elastycznych (2). A.Pluta. 3/86 22.
Pamięci na cenzurowanym. J.Orkiszewski. 2/86 15.
Płyta główna PC XT. S.Piotrowski. 7/87 15.
Płyta główna PC XT. Z.Kądzielski. 8/87 15.
Porównanie architektury 16-bitowych mikroprocesorów firm Intel i Motorola czyli czemu wolę Motorolę. A.Forycki. 2/87 11.
Przestrzeń adresowa IBM PC. Z.Pojmański. 6/87 16.
RS-232C Interfejs szeregowy w IBM PC XT/AT. S.Piotrowski. 6/87 24.
Sieć w sieci. J.Wrześniak. 8/87 30.
Standard Centronics. A.J.Piotrowski. 2/86 21.
Trzy i pół calowe niemowlę. 1/86 6.

Systemy operacyjne

Czym jest UNIX. J.Zalewski. 1/86 21, 2/86 18, 3/86 8, 1/87 27. R.Grabowski, J.Zalewski. 2/87 22.
DOS 3.3 7/87 2.
Dusza komputera 4/87 23.
GEM system operacyjny dla każdego. R.Liwiński. 3/87 10.
Jeden dla wszystkich. 1/86 11.
MS-DOS i jego korzenie. 3/87 18.
MS-DOS od środka. Z.Pojmański. 5/87 11, 6/87 8, 7/87 20, 8/87 12, 9/87 23.
MS-DOS symuluje CP/M. 6/87 31.
Następca MS-DOS. Z.Pojmański. 7/87 4.
OS/2 - nowy system operacyjny dla IBM. 5/87 29.
OS/2 i bazy danych. 7/87 2.
System operacyjny dla 80386. 6/87 19.
Ukryte gwiazdy. 2/86 23.
UNIX w drodze po sukces. 2/86 20.

Tajniki programowania

BIOS IBM PC. Z.Guzik. 9/87 15.
Elementy grafiki w języku Turbo Pascal. J.Bielecki. 1/87 30.
Język C zarządzanie pamięcią operacyjną. J.Bielecki 3/87 2.
RS-232C Wykorzystanie procedur BIOS. A.J.Piotrowski. 6/87 27.
Z .EXE na .COM. M.Mizeracki. 6/87 11.
Zabezpieczanie programów. J.Kroh. 9/87 19.

Technologie i podzespoły

17 MIPS. 5/87 29.
Dyski optyczne. 6/87 30.
Kres scalania układów? 6/87 23.
Łączniki. 9/87 26.
Na spiralnej ścieżce. 8/87 6.
Najszybszy układ świata. 6/87 6.
Nowe kostki. 1/86 27, 2/86 14.
Nowy mikroprocesor Motoroli. 6/87 23.
Rodzina 80386 poszerza się. 5/87 29.
Transputer już na rynku. 2/86 31.
Układ na zamówienie. 6/87 19.
Układy biologiczne. 2/87 15.
Układy pamięci 1 MB. 6/87 19.
Zmierzch standardu TTL? 6/87 29.

Testy mikroklanu

BIM PC AT. A.J.Piotrowski. 7/87 23.
Epson LX-800. S.Piotrowski, A.J.Piotrowski. 8/87 23.
Regeneracja kasetek do drukarek. M.A.Markowski. 8/87 26.

Zastosowania

Co to jest CAD/CAM? 3/86 10.
Druga młodość telefonu. A.Suwała. 6/87 20.
Głupi jak but? 8/87 22.
Komputer czy cyrkiel i linijka? Barbara Chrzan-Feluch. 3/86 28.
Komputer kierowcą. 6/87 29.
LAN i światłowody. 9/87 14.
Luksus czy pożyteczne narzędzie? 2/86 8.
Mikrokomputer zamiast deski kreślarskiej. 3/86 30.
Mowa syntetyczna. R.Ciarkowski. 9/87 7.
Możliwości sieciowe. 7/87 3.
Obsługa baz danych. R.Szuniewicz. 8/87 2.
PC XT/AT zamiast terminalu. S.Dramiński. 7/87 31.
Pierwsze kroki. 1/86 4.
Płytka z komputera. Adam Kutszyński. 9/87 26.
Skok po kroku. K.Kamiński. 6/87 12.
Symulacja drgań w dwóch wymiarach. B.P.Zurawski 2/86 29.
Teletex. A.Suwała. 9/87 28.
Trzeci wymiar. R.K.Mołdach. 9/87 29.
Wielodostępne bazy danych. A.Suwała. 8/87 5.

Różne

Ankieta. 3/87 15. Wyniki ankiety 8/87 32.
Bit za trzy grosze. J.Tatarkiwicz. 1/86 28.
Bit, Bajt i Spółka. 2/86 4.
Ceny. 1/86 32, 2/86 32, 7/87 14.
Czy SAA zlikwiduje antagonizmy? A.Suwała. 7/87 13.
Dwunożny robot. 6/87 6.
Elementarz dla profesjonalistów. M.A.Markowski. 9/87 31.
Jak nas piszą... tak nas widzą. T.Kuczborski. 6/87 II.
Jak pisać do **mikro**klanu. 1/86 32, 2/86 32, 3/86 III, 1/87 32.
Komputerowa obsługa wizyty Papieża. 5/87 10.
Krzywa smoka. W.Guziński. 6/87 30.
Między Londynem i Warszawą. 3/87 III.
Nie tylko dla profesjonalistów. A.J.Piotrowski. 5/87 II.
Nowa magistrala. 6/87 30.
Od Warszawy do Wrocławia J.Rudomina. 5/87 32.
SAA - nowa koncepcja rozwoju IBM. 5/87 7.
Spektroskopia Spectrum. B.Mank. 2/87 III.
Targi Poznańskie. 5/87 III.
Nie tylko dla profesjonalistów. 6/87 IV.
W królestwie Fraktali. Z.Pojmański. 8/87 6.

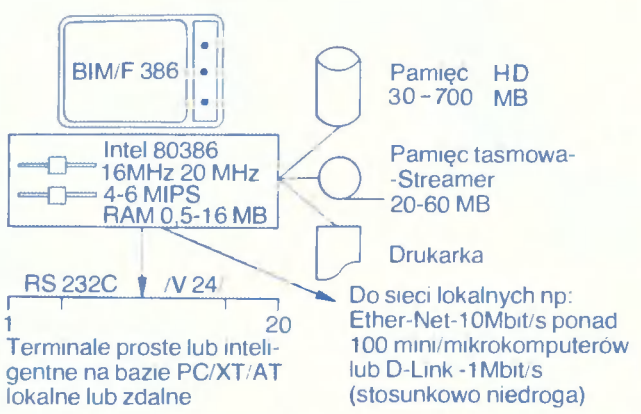
PZ globo®

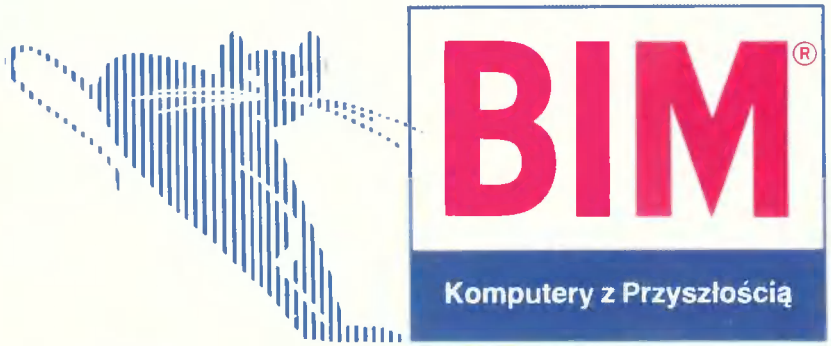
Licencjonowany producent firmy BIM
Koplin 73 200 Choszczno Tel. 7550 Telex 0445413

Oferuje niezawodne, bardzo wydajne, nowoczesne:

- Komputery 32-bitowe **BIM /F 386** – kompatybilne z **IBM PC/AT**
- Komputery 16-bitowe **BIM PC/XT/AT** – kompatybilne z **IBM PC/XT/AT**
- Terminale, modemy, koncentratory transmisji danych.
- Systemy użytkowe: **F-K,** materiałowy, kadrowo-płacowy, kosztorysowania.
- Sieci komputerowe (Ether-Net, D-Link) oraz systemy wielodostępne na bazie **BIM /F 386, BIM PC/XT/AT** realizowane „pod klucz" – z oprogra–mowaniem systemowym i użytkowym, terminalami, modemami z instalo–waniem i szkoleniem

Przykład sieci opartej o komputer 32-bitowy BIM/F 386





BIM jest zastrzeżonym znakiem towarowym firm BIM Technologies AG i PZ Globo
IBM jest zastrzeżonym znakiem towarowym International Business Machines Corporation

W mikroklanie za miesiąc

Desktop Publishing Gwiazda czy meteor?